SINCLAIR
QL
WORLD
JANUARY 1992 £1.95
CATCH JACK
I.D. game
to type in
A QUESTION
OF DOTS
Old and new
DTP programs
NEW
APPLICATIONS
FOR ABACUS
M.C.M.
QUALITY
EDITORIAL
01
9 770951 933016

DJC

Dilwyn Jones Computing

41 Bro Emrys, Tal-y-Bont,
Bangor, Gwynedd LL57 3YT
Tel: Bangor (0248) 354023

WE WOULD LIKE TO WISH OUR CUSTOMERS A MERRY CHRISTMAS AND A HAPPY, PROSPEROUS NEW YEAR.

SEND SAE OR IRC FOR A COPY OF OUR CATALOGUE CONTAINING FURTHER DETAILS OF OUR RANGE OF SOFTWARE

SOFTWARE DISCOUNTS

Buy 2 programs together CLAIM 5% OFF
Buy 3 or more programs CLAIM 10% OFF
Vision Mixer Plus and PictureMaster packages described below count as ONE program for this discount offer.

PROGS (Professional & Graphical Software)
THE PAINTER £25.00
Superb painting program for the QL (512k)
THE CLIPART £12.00
3 disks full of QL clipart. (128k)

DATA DESIGN version 2.03 £50.00
Programmable database, can be programmed from SuperBASIC via new easier to use BASIC extensions. (512k)
QRACTAL £20.00
100% machine code Fractals program (512k)

ROB ROY QL BARGAIN SOFTWARE PACK
On disk only £10.00
on 3 microdrive cartridges £14.00
INKWELL DELUXE - 100's of print styles from Epson compatible 9 pin dot matrix printers. Font editor + 20 extra fonts
INKWELL DELUXE TYPER - Typewriter style extension to Inkwell Deluxe
CUEWELL 1 + CUEWELL 2 - front end utilities

ADDRESS LABEL PRINTER *by J.H. Ree* £15.00
On microdrive cartridge £15.00
(384k, RAMdisk required)
100 ADDRESS LABELS on printer roll for use with Address Label Printer £2.00
add 50p postage per roll (max £2.50)

REMIND ME *by Joe Haftke* £12.00
On microdrive cartridge £12.00
Dates reminder program (128k)

QL GENEALOGIST *by Chris Boutal*

See our other advert elsewhere in this issue for details of the latest version of this family history program.

SUPER DISK LABELLER £10.00
On cartridges if required £12.00
Print labels for floppy disks, can show the filenames on disks in columns. (256k)
ROLL 100 DISK LABELS for above £2.50
add 50p postage per roll of labels

SUPER DISK INDEX *Iby mre Dominik* £12.00
On microdrive cartridge £12.00
Catalogue and index your disks as a database, with search, sort, print, etc. facilities. (256k)

DISCOVER *by Dave Walker* £20.00
QL/PC/ST disk file transfer (256k, disk only)

MULTI DISCOVER *by Dave Walker* £30.00
As Discover, also transfers to BBC DFS/ADFS, CPM (many formats) and UNIX CPIO formats (256k, disk only)

TEXTIDY *by Dave Walker* £15.00
Text file conversion utility, e.g. for use with Discover above. (256k, disk only)

QUICK POSTERS *by Dilwyn Jones* £10.00
on 2 microdrive cartridges £12.00
Star printers text poster maker (128k)

BASIC REPORTER *by Dilwyn Jones* £10.00
on microdrive cartridge £12.00
Super BASIC programming aid (128k)

SCREEN ECONOMISER *by G. Estournet* £10.00
on microdrive cartridge £12.00
Screen protection and blanking utility - turns off the QL display if no key is pressed for set number of minutes.(128k)

TRANS 24 *by Ralf Reköndt* £10.00
on microdrive cartridge £12.00
9 - 24 pin printer graphics dumps conversion utility (128k)

FILES 2 *by Peter Jefferies* £12.00
on microdrive cartridge £12.00
Comprehensive popup files utility. (128k)

SPELLBOUND *by Peter Jefferies* £30.00
on microdrive cartridge £30.00
Check spelling as you type, 30,000 word dictionary, expandable by the user. (256k)
SPELLBOUND SPECIAL EDITION £50.00
not available on microdrive cartridge
Larger dictionary, turns back on automatically at start of new words. (512k)
UPGRADE, SPELLBOUND TO SPELLBOUND S.E. £30.00
Return Spellbound master disk/cartridge

FLASHBACK *by Peter Jefferies* £25.00
on microdrive cartridge £25.00
Excellent fast, versatile database (256k)
FLASHBACK SPECIAL EDITION £40.00
not available on microdrive cartridge
Enhanced version of Flashback - new commands, report generator, etc. (256k)

TASKMASTER *by Peter Jefferies* £25.00
on microdrive cartridge £25.00
Front end utility for the QL (384k)
Hard disk users - ask about the version of Taskmaster for hard disks

WINBACK *by Norman Dunbar* £25.00
Miracle hard disk backup utility. (256k, Toolkit 2, available on disk only)

THE GOPHER *by Norman Dunbar* £12.00
on microdrive cartridge £12.00
File finder utility. WIN or FLP (128k)

QUESTION MASTER *by C.B. Storey* £10.00
on 2 microdrive cartridges £12.00
Question/answers quiz and revision system. Quiz editor included (128k)
General Knowledge quiz £5.00
Classical music quiz £5.00
QLestions - quiz on the QL itself! £5.00

FILEMASTER *by Joe Haftke* £12.00
on microdrive cartridge £12.00
File care utility, directory display in columns, disk labelling. (512k, RAMdisk)

HOME BUDGET *by Joe Haftke* £20.00
on microdrive cartridge £20.00
Personal income tax calculator (UK) plus domestic bills and accounts system (128k)

VISION MIXER 1 *by Dilwyn Jones* £10.00
on 2 microdrive cartridges £12.00
Screen pictures video effects display, as reviewed in QL World Jan 1991 (384k)
VISION MIXER PLUS *by Joe Haftke* £22.50
available on disk only
More features than version 1, even easier to use. (256k)
Upgrade VISION MIXER 1 TO VISION MIXER PLUS - *(quote serial number)* £14.50
PICTUREMASTER *by Joe Haftke* £15.00
available on disk only (256k)
Screen making utility for Vision Mixers.
VISION MIXER PLUS + PICTUREMASTER - both together, save 20% £30.00
PICTUREMASTER + the upgrade £24.00

COCKTAILS WAITER *by Imre Dominik* £10.00
on 2 microdrive cartridges £12.00
Cocktail drinks recipes (over 400 supplied!) database. (256k)
MIX 2 - hundreds of extra recipes £5.00
MIX 3 - even more recipes, includes many non-alcoholic recipes £5.00
MIX 2 or 3 on microdrive cartridge £6.00

DI-REN PRODUCTS

FLEET TACTICAL COMMAND II £49.95
on microdrive cartridge £56.95
DATA PRINT UTILITY for FTC II £9.95
FLEET TACTICAL COMMAND I £29.95
THE FUGITIVE text adventure £9.95
QL NETWORK PROVER hardware £3.50
MINI PROCESS CONTROLLER hardware £59.95
6 relay switches, controlled by simple commands via the QL serial port.

MPC UTILITY software £9.95
Control program for process controller. Includes tookit of BASIC extensions for writing your own controller software.
Leads and PSU's for the MPC also stocked.

Send SAE or international reply coupon for a copy of the DI-REN catalogue.

SUNDRY SUPPLIES

Brand new microdrive cartridges	£2.50	100 disk labels on printer roll	£2.50
Roll 100 cartridge labels	£2.00	100 address labels on roll, 3.5" wide	£2.00
3.5" DSDD unbranded disks + labels	£0.45	Disk storage boxes for 3.5" disks:	
20 or more	£0.40	10 disks (Slimpak boxes)	£1.20
3.5" Sony branded disks + labels	£1.00	40 disks, clear, lockable hinged lid	£5.00
pack of 10	£9.00	80 disks, clear locakable hinged lid	£7.00
5.25" DSDD unbranded disks + labels	£0.35	160 disks, POSSO box, drawer style	£17.00
Extra disk labels, roll 100	£2.00	Disk box subject dividers, per set of 20	£3.00
		Monitor stand	£15.50

SOFTWARE POSTAGE: Software sent post-free to UK addresses. Abroad add £1.00 per program for postage and packing. Supplies postage - add £2.50 per order to UK addresses (50p per roll of labels, see above). Abroad, add standard rate of £3.50 for supplies postage, except for airmail outside Europe, where you should add 10% of value of order, minimum £3.50. PAYMENT - We can accept payment by cash, cheque, Eurocheque, postal order, international postal/money order in British currency (pounds Sterling) only please. We can now also accept credit card payments using the following cards: VISA, ACCESS, MASTERCARD, or EUROCARD. Please state your card type, number, expiry date, address (goods must be sent to address of card holder) and remember to sign your order. Goods remain property of DJC until paid for in full.

VISA Access EUROCARD Master Card

SINCLAIR QL WORLD

Editor
Helen Armstrong

Production Controller
Jayne Penfold

Designer
Jeff Gurney

Advertising Manager
Jason Newman

Magazine Services
Yvonne Taylor

Advertising Production
Michelle Evans

Group Advertising Manager
Lynda Elliott

Group Editor
John Taylor

Publishing Director
Wendy Palmer

Group Deputy Managing Director
Ray Lewis

Group Managing Director
Peter Welham

Sinclair QL World
Panini House
116-120 Goswell Road
London EC1V 7QD
Telephone 071-490 7161
ISSN 026806X

Unfortunately, we are no longer able to answer enquiries made by telephone. If you have any comments or difficulties, please write the The Editor, Open Channel, Trouble Shooter, or Psion Solutions. We will do our best to deal with your problem in the magazine, though we cannot guarantee individual replies.
Back issues are available from the publisher price £2 U.K., £2.75 Europe. Overseas rates on request.
Published by Maxwell Consumer Magazines, A Division of MCPC Ltd., Sinclair QL World is distributed by IPC Market force, King's Reach Tower, Stamford Street, London SE1 9LS.
Subscription information from: MCM Subscription Dept, Lazahold Ltd., PO Box 10, Roper St, Pallion Ind. Est., Sunderland SR4 4SN. Tel: 091 510 2290
UK: £21.00, Europe: £32.00,
Rest of the World: £38.00.
Please include your subscriber number with any queries.

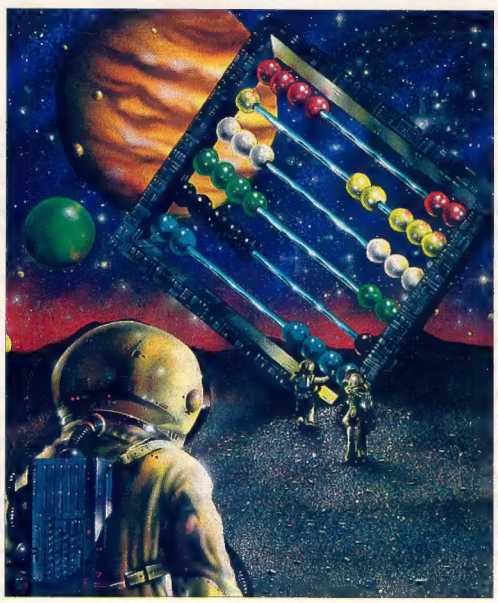
Typesetting by Ford Graphics, 8-10 Whitsbury Road, Fordingbridge, Hampshire. SP6 1BR. Tel: (0425) 655657
Printed and bound by BPCC, Colchester. Covers printed by Spottiswoode Ballantyne, Colchester.
Sinclair QL World is published on the third Thursday preceding cover date.

© COPYRIGHT
SINCLAIR QL WORLD – 1991

CONTENTS

JANUARY 1992

4 TROUBLESHOOTER ● WP possibilities and front ends
11 NOTICEBOARD ● Forward planning
13 QL SCENE ● Special offer from Care
14 OPEN CHANNEL ● Backup dropouts
16 A QUESTION OF DOTS ● The Word Processor's Progress
22 NEW USES FOR ABACUS ● Not just a spreadsheet . . .
25 THE NEW USER GUIDE ● Part 11 BEEP to CLOSE
29 SOFTWARE FILE ● Squidgy
33 DIY TOOLKIT ● Front ends and name tables
35 SOFTWARE FILE ● Qubbesoft Public Domain
36 PROGS! ● Catch Jack
42 SYSTEMATIC MACHINE CODE PROGRAMMING ● Part 3
46 SUBSCRIPTION INFORMATION
47 BOOK REVIEWS ● Old and new books across the board



NEXT MONTH

We are still restraining a review of PROSPERO PASCAL for programmers, a ONE MAN's SYSTEM about Archive, lots of reviews including several low-cost utilities, and a user report on the CV1 digitiser, as well as machine code programming ideas from DIY Toolkit and Alan Bridewell.

TROUBLE

A PROBLE

Bryan Davies assesses the advanced state of GL word processors and front ends and asks: what next?

With *Perfection* now well-known, and an upgraded *text*[87] introduced, it may not seem the obvious time to ask for comments on what you feel is still missing from the available wp programs, but it is desirable to 'get your oar in' early if you want to influence future developments of wp programs for the QL. There is still much improvement to be made in wp programs for *any* microcomputer. It may be that wp and dtp will gradually merge; wp programs are certainly moving closer and closer to dtp programs in their capabilities. Like it or not, the bigger market outside the QL scene will tend to set the pace for us, and we can expect the introduction of such features as wimp interfaces (provided enough QL users keep on buying software to make it worthwhile suppliers staying in the market).

Graphical interfaces

There seems little doubt that the Apple Macintosh is the leader in the use of graphical interfaces. Running to catch up are the Microsoft Windows-based programs on the PC. My own recent experience with *Windows* and the matching wp program *Word For Windows* is that they both leave a great deal to be desired (as does the technical support). This does not alter the fact that these programs point the way *we* are likely to go (ie be taken); it is just that they don't (to my mind) make a good job of helping the user get on with work. There is a considerable emphasis on gimmicks – interesting, but no use if you have work to do. This may be a reflection of life in the 1980s and 90s in general; we have to have pretty pictures and buttons to press, even if this means it takes longer and more effort to perform a desired function.

In case wimp and gui don't mean anything to you, here's a brief summary of their meaning and intent. They are essentially the same thing. Wimp stands for windows, icons, mouse and pull-down menus, gui for graphical user interface.

Different programs, or different parts of the same file, can be displayed running in separate areas of the screen, simultaneously. Programs, files and actions are displayed in the form of small pictorial representations of themselves, such as a square, white object with horizontal black lines across it to indicate a text document (file). Basic menu names are shown on single bars, typically across the top of the screen, and the options for a menu are displayed on a panel which 'drops down' (or is 'pulled-down') if the menu name is selected. The mouse steers a pointer; selecting a menu or option is done by moving the pointer onto the icon for the menu or option and pressing a mouse button to select it. Programs are started by pressing the mouse button twice ('double-clicking') when the pointer is on the program icon. Rather than typing text in on a command line, as with Basic, the idea is to use a mouse to point-and-click on clearly-identified areas of the screen, to obtain the same functions. A simple example is deletion of a file; instead of typing-in the delete command and the full device, path and file name, then pressing ENTER, you move the mouse until its pointer lies on the icon identifying the file concerned, click a mouse button to 'mark' that file, then move the pointer to the delete option on a pull-down menu and click again. Better still, with the Mac and some other micros, you point to the file and hold the mouse button down while 'dragging' the file to a rubbish bin (trash can) icon, then release the button with the pointer on the bin to delete the file.

Easy access

The aim is to make hard-to-remember commands easy to access. This will cut no ice with hackers who pride themselves on knowing all the Basic and program commands, and use them often enough not to forget them, but many users have neither the ability nor the interest to learn commands which are often less than clear. Evidently, we are – to some extent – being dragged along a path that is being created primarily for wp operators in offices. However, many home micro users will be quite happy to have their computer life made easier for them this way.

While the mouse has been an integral part of the graphical interface all along, the user is normally given the option of selecting operations from the keyboard. In general, selection is quicker by key than by mouse. A mouse sounds a great idea until you actually use one in earnest; it then tends to feel distinctly cumbersome. For a fast typist, the mouse can be a deadly enemy, since its use entails removing a hand from the keyboard, thereby destroying the rhythm of typing; you can hardly keep up 120 words per minute typing speed with one hand periodically being 'lost'. To be fair to the mouse, the flow of typing is interrupted anyway whenever a command has to be selected from the keyboard. Whether or not the user takes to the mouse depends to a large extent upon how easily and quickly functions can be accomplished by mouse, and how memorable the alternative keyboard actions are. To return to Windows and Word For Windows, the former has plenty of quite forgettable key combinations, whereas the latter has several eminently sensible ones. There's no great problem remembering Ctrl-B to switch on or off the Bold text attribute, but what about Alt-down cursor to open a list of options in a menu window, then up/down cursor to identify the option required, then Alt-up/down cursor to select that option? Depending on the type of menu displayed, the key combinations vary, just to make life more confusing. My own reaction was to immediately abandon any thought of using the keyboard for the one program, but to use it for the other; not exactly a consistent user interface.

Well managed

On the QL, Qpac is the equivalent of Windows, having quite a lot in common with it. Qram or Qpac users will be familiar with many mouse actions, although some dragging and resizing functions may not be available to them in the most intuitive form. Both programs are Managers, definitely with a capital 'M'. They effectively take control of the computer, and other programs and users have to live by their rules. Ideally, from the point of view of Qpac, applications programs should be written (or rewritten) to be compatible with – and make full use of – Qpac. To avoid extensive keying, a mouse is virtually essential, and we don't have a 'common mouse' for the QL. The Quanta effort to

SHOOTER

MSOLVED

produce its own Qimi mouse interface, with some software improvements, suffered some delays, and Qimi is not compatible with many existing (and important) application programs anyway. We shouldn't forget *Ice* either, as that was really the first useful, and usable, wimp interface for the QL; a great pity it was not developed to provide other functions, such as wild-card copying, dragging icons for copying and deletion, double-clicking on files to automatically start the program they were created in and load the files etc.

Returning to the initial point in this article, if you have strong feelings on how both programs and the user interface should be developed, let us know.

Fancy stuff?

Users of Quill may say 'why bother with all this fancy stuff?', and I tend to agree, *if* you use (and are satisfied by) just that one program. The limited facilities it provides are obtainable without too much pain from simple keyings. By-and-large, the other Psion programs use the same keyings, making it unnecessary to remember different combinations for similar commands in each program. There would not have been such a large variety of alternative programs produced for the QL if *Quill* had been thought satisfactory by everyone, though. The demand for additional functions in individual programs, and for the ability to have several programs available quickly, at the touch of a couple of keys, has brought us to a much higher state of software development. Serious users will not want to stop there, either. The more non-Psion programs one uses, the greater becomes the headache from trying to remember the key combinations for commands. This is one problem that the Windows interface has tried to overcome; the idea is that *all* programs will utilise the *same* menu structure. When you start any program, it will have essentially the same menu bar across the top of the screen, and most of the same options will appear on the drop-down menus. The user is, therefore, able to use *any* program – even one newly-installed – without having to read-up on key combinations for commands. Surprisingly enough, it works quite well in practice. Can we reasonably hope for something like it on the QL, one day?

Graphical user interfaces suffer from being slower than text ones, but the Gold Card and HD/ED floppy drives or hard disk have now given the QL the necessary speed to handle the graphics operations better, and the non-graphics operations much better.

Readers' letters

David Cottom enquired about a problem he had printing from *text*[87] to an Epson GQ-5000 laser printer. The problem was that text was not being wrapped at line-ends. As he uses a serial-to-serial link from QL to GQ, his system is not quite comparable with mine, on which the serial port output is converted by a Miracle serial-parallel interface and fed into the parallel input on the printer. One reason for *not* using the serial input is the relative complication of setting it up; you have to set word length, Baud rate, parity, stop bit, DTR, XON/XOFF, DSR, CTS, and RX buffer. Don't ask me to explain all these: with the parallel input, all you do is plug the cable in. There is another difficulty with using the serial input, in that you will need a cable with the QL PCC connector on one end and a standard printer 25-pin 'D' connector on the other. Although there are 25 pins on the printer connector, the QL SER1 and SER2 ports use only four wires, but not the same four. The only suggestion I could make to Cottom here was that the CTS function should be set Off on the printer, as the QL does not use that line for SER1.

> *"We can expect features like wimp interfaces, as long as QL users buy enough software ... many users will be happy to have their life made easier in this way."*

The more likely place to look for the failure to wrap lines is in text[87] itself. The original GQ driver worked well *unless* you set the Layout to get the maximum left and right margins; to the best of my recollection, setting the left margin as close to the edge of the paper as possible resulted in no line wrap, but you could fix this by moving the left margin even one keypress to the right. This problem should not occur with later versions of the driver.

A. Ingrey ordered a keyboard from **Keyboard Products** and managed to get a refund for it after about four months (this was one of several orders we received complaints about and chased-up). He then ordered another keyboard from **EEC Ltd** in November 1990, and that appeared to get lost in the post; as of July 1991, he had received neither keyboard nor refund, although the latter was apparently being sought from the Post Office by EEC. So far, no reply has been received to my enquiry, from EEC.

Lost information

T.K. Computerware has had a hard disk problem recently *(not* with a QL unit, I believe) and this has caused some difficulty tracing information on orders, but they have supplied the following comments on letters sent to us. **A.P. Campbell** ordered the game *Patience* in June '91 and wrote in July to say it had not arrived; T.K. had despatched the order, but it went to the wrong Campbell. The matter has now been sorted out. **N.D. Mortier** ordered a book and some cartridges in May '91 and got the book in June but had not received the cartridges as of mid-August. T.K. report sending the cartridges and receiving them back from the Post Office marked 'not called for'; they were sent to Mortier again, and he should have received them by now. Presumably the UK Post Office now has a standard policy of calling only once to deliver anything; if you are not in to receive items which will not go through the letter flap, or require a signature to confirm receipt, the postperson should leave a notice advising you that the call has been made and giving details of where to go to collect the goods.

Chess checked

A.R. Kempton ordered the Psion game *Chess* in May '91. It was apparently despatched but lost in transit. While it was being looked for, Psion decided not to produce any more Chess cartridges, partly because of the difficulty they were having getting cartridges and partly because there were too few orders to justify continuing the product. T.K. therefore refunded Kempton's money. **Dirk van Rompuy** ordered an EPROM eraser and the ST-Z88 link software in March '91. The order was

despatched in two parts, in early April and early May. As of June, Rompuy said he had received only the eraser. Hopefully, the software has also arrived by now.

A few suggestions on how to avoid some of the complaints made about suppliers: when leaving messages, make sure to leave an office 'phone contact number as well as a home one (if you have both); make sure the expiry date you give with credit card details is correct (card companies can turn down authorisation requests if even small account details are incorrect); send valuable returned goods back *insured* and by a confirmed-delivery service. It may be that some buyers are receiving less speedy service now than they did a year or so back, from the same supplier, and this really should not come as a surprise. Consider the general state of the UK economy (also those of many other countries) and ask yourself what most stockists of goods will have done in the past couple of years; for certain, they will have reduced stock levels, to keep their cash outlay down. Only when orders are received will they be placing orders with their suppliers, and the latter also are likely to keep less stock than before. Delays are sometimes going to occur right down the chain.

Another area where delay can occur is the gap between receipt of a cheque and despatch of the ordered goods. Whereas some suppliers might, in the past, have sent goods as soon as the cheque was received, they will (if they are sensible) now be waiting long enough after a cheque is banked to ensure that it does not 'bounce'. Banks claim, nominally, that cheques are cleared in two to three working days, but this cannot be relied upon. One bank I deal with does not credit *any* cheque in under seven working days; that means a maximum of nearly twelve days before you can use the money if you make a deposit late on a Friday. A supplier cannot safely consider a cheque has cleared two to three days after it has been banked; it might re-appear as 'returned' some time late on the third day, or even on subsequent days. Banks are now charging far more for their (sometimes dubious) services than they used to do, and returned cheques are costly ones. A point to bear in mind regarding payment by credit card is that the total cost of single orders has to be charged as one transaction, even though the goods may be sent in more than one package, at different times. This is the way the card companies require the supplier to do business. It may mean you get part of an order but are charged for the whole order; if there is then some problem with shipment of the rest of the order, you stand to be out of pocket for *that part* of the order until checks have been made on the missing item(s).

"There are less complaints about suppliers than there were ... it's heartening to see letters containing technical queries."

New owners call

There are many less complaints about suppliers than there were even a year ago. Although the general drop-off in QL business, and the recession, must have been major factors in this change, it is heartening to see most letters containing technical queries rather than complaints now. For one reason or another, the less reliable suppliers have largely left the scene. A significant portion of current letters appears to come from new QL owners, who often ask questions that have been answered in these pages several times before. Older readers should bear in mind the problems *they* had getting information, years ago, when tempted to object to repeated bits of advice! There are less sources of help now than there were then and we feel it is our job to assist the new user. It is to the benefit of all of us to have more users buying more products and helping to keep the QL world healthy.

C.G.H. SERVICES

Cwm Gwen Hall, Pencader, Dyfed, Cymru, SA39 9HA (Tel. 0559 384574) (9am - 5pm)

COMMERCIAL SOFTWARE

ANELPUM QUAT (NICK WARD) (text adventure) (128k) (flp/mdv) £10.00
ASSAULT AND BATTERY (DAMON CHAPLIN) (shoot 'em up) (128k) (flp/mdv) £12.50
ASTRO (NICK WARD) (astrology prog) (128k) (flp/mdv) £12.50
THE BLAG2 (TONY WOOLCOCK) (text adventure) (256k) (flp/mdv) £10.00
D-DAY MKII (DDC & RICH MELLOR) wargame) (256k) (flp) £15.00
DOUBLE BLOCK (FRANCOIS LANCIAULT) (arcade game) (128k) (flp/mdv) £10.00
DREAMLANDS (JEAN-YVES ROUFFIAC) (text adventure) (256k) (flp) £10.00
EPIC ADVENTURE (ANDY PRITCHARD) (illustrated adventure) (256k) (flp) £12.50
FIVE GAMES PACK 1 (WREFORD DAVIES) (mind games) (128k) (flp) £12.50
FRACTALS FROM NEWTONS METHOD (JOHN TOPHAM (fractals) (128k) (flp/mdv) £12.50
FROM THE TOWER OF VALAGON (ALAN PEMBERTON) (text adventure) (128k) (flp/mdv) £10.00
GEE-GEE SYSTEM (PHIL JONES AND ANDY CSERBAKOI) (race predictor) (128k) (flp/mdv) £10.00
GREY WOLF (OLIVER NEEF) (u-boat simulator) (256k) (flp) £10.00
HERE WE GO (PHIL JONES AND ANDY CSERBAKOI) (text adventure) (128k) (flp/mdv) £10.00
MACSPORRAN'S LAMENT (DAVE WATSON) (illustrated text adventure) (128K) (flp/mdv) £10.00
OPEN GOLF (OLIVER NEEF) (golf simulator) (384k) (flp) £12.50
ORBITING STARS (JOHN TOPHAM) (astronomical simulator) (128/256k) (flp/mdv) £10.00
PERSONAL FINANCE MANAGER (JASON VICINANZA) (budget system) (128k) (flp/mdv) £10.00
POLYTEXT (NICK WARD) (multi-column quill) (128/256k) (flp/mdv) £17.50
PUDGE (DAMON CHAPLIN) (arcade game) (128k) (flp/mdv) £12.50
QUICK MANDELBROT (KENNETH MURRAY) (fractals) (128k) (flp/mdv) £10.00
QUICK MANDELBROT & JULIA (KENNETH MURRAY) (fractals) (128k) (flp/mdv) £12.50
QUIZMASTER (PHIL JONES AND ANDY CSERBAKOI) (quiz game) (128k) (flp/mdv) £10.00
QUIZMASTER II (AS ABOVE & RICH MELLOR) (quiz game + module option) (128k) (flp/mdv) £12.50
RETURN TO EDEN (OLIVER NEEF) (role-playing illustrated adventure) (256k) (flp) £17.50
SCRIPTWRITER (ANDY PRITCHARD) (text processor) (256k) (flp/mdv) £15.00
SECTOR X (HORST SPIERLING) (shoot 'em up) (256k) (flp) £12.50 (mdv) £15.00
SPEEDFREAKS (DAMON CHAPLIN) (car race game) (128k) (flp/mdv) £12.50
SQUIDGY ROUND THE WORLD (MICHAEL CROWE) (arcade game) (flp) £12.50 (mdv) £15.00
STARPLOD (ALAN PEMBERTON) (icon-driven adventure) (128k) (flp/mdv) £10.00
3D TERRAIN (JAN THOMPSON) (use abacus data to create 3D graphs) (128k) (flp/mdv) £12.50
STOQL (ALAN PEMBERTON & RICH MELLOR) ST 2QL) (screen transfer & image processor) (256k) (flp/mdv) £10.00
UNCLE LOONIE'S LEGACY (DAVE WATSON) (puzzle adventure) (128k) (flp/mdv) £10.00
VOYAGE OF THE BEANO (ALAN PEMBERTON) (illustrated text adventure) (256k) (flp) £12.50
WRECK DIVE (NICK WARD) (arcade adventure) (128k) (flp/mdv) £10.00

CLIP ART 1 - SPORTS (FLP) £6.00
CLIP ART 2 - WHIMSIES (FLP) £6.00
CLIP ART 3 - OFFICE (FLP) £6.00
CLIP ART 4 - VIZ (FLP) £6.00
CLIP ART 5 - GENERAL (FLP) £6.00

SPEEDSCREEN - ROM VERSION £20.00
SPEEDSCREEN - FLP/MDV £10.00

PUBLICATIONS

QL TECHNICAL REVIEW ISSUES 1-2 £1.50 EACH
QL TECHNICAL REVIEW ISSUES 3-6 £1.75 EACH
QL ADVENTURERS FORUM ISSUES 1-3 £1.25 EACH
QL ADVENTURERS FORUM ISSUES 4-9 £1.50 EACH
QL LEISURE REVIEW ISSUE 1 £1.75 EACH
INTERNATIONAL QL REPORT ISSUES 1-2 £1.50 EACH
QL SURVIVOR'S SOURCE BOOK £2.50 EACH

PUBLIC DOMAIN AND SHAREWARE LIBRARY
ALL DISKS £2.00 EACH INCLUSIVE OF MEDIA AND P&P

ADVENTURE GAMES DISK 1 (includes fantasia and ye classical type adventure)
ADVENTURE SOLUTION DISKS 1-3 (includes The Pawn and Mortville Manor, mainly ST thought)
ADVENTURE SOLUTIONS DISK 4 (QL specific)
ADVENTURE GAMES SOURCE CODE DISK 1 (includes Fantasia, Haunted House, etc)
ADVENTURE UTILITIES DISK (includes Quill to SuperBasic converter and demo adventure)
AUSTRALIAN P.D. DISKS 1 + 2 (includes a wide variety of games, utilities etc)
COMMUNICATIONS DISK 1 (includes QBOX Bulletin Board system)
COMMUNICATIONS DISK 2 (includes QL Kermit)
CONNECTIONS DISK 1 (includes ST - QL screen and file converters)
DEVICE UTILITIES DISK 1 (includes Archiving, compacting, formatting and Shell programs)
DILWYN JONES DISK 1 (includes many SuperBasic progs and Wordsearch)
EDITORS DISK 1 (includes QED and MicroEmacs)
EDUCATIONAL PROGRAMS DISK 1 (includes maths, music and chemistry programs)
EMMANUEL VERBEECK DISK 1 (includes screen save and print progs. and many more utilities)
ESOTERICA DISK 1 (includes DIY Pyramid construction prog. Biorythms and Psychology progs)
FRACTALS DISK 1 (includes large numbers of mandelbrot and other fractal progs)
FRACTALS DISK 2 (Carl Cronin's mandelbrot prog plus animation screens)
FRACTALS DISK 3 (Rainer Kowallick's mandelbrot prog as amended to give "Jewel" effect)
GAMES DISK 1 (includes Starburst, Cavern Frenzy)
GAMES DISK 2 (includes many arcade type games)
GAMES DISK 3 (includes QL War)
GERMAN TO ENGLISH DICTIONARY (multi-tasking)
GRAPHIX ANIMATION DISKS 1, 2, 3, 4, 5 (TV Movies)
GRAPHIX DEMOS DISK 1, 3, 4
GRAPHIX SCREENS (FIF FORMAT) DISKS 1, 2, 3
GRAPHIX SCREENS (PIX FORMAT) DISK 1
GRAPHIX SCREENS (SPECTRUM) DISK 1
GRAPHIX SCREENS (ST) DISKS 1, 2, 3, 4, 5, 6, 7, 8
GRAPHICS/SCREEN UTILITIES DISK 1 (includes sprite designer. CAD and window progs.)
HAM RADIO DISK 1 (includes PC conversions)
INDEXES DISK 1 (includes calculator, pi and calendar creator progs.)
OLIVER FINK DISK 1 (includes progs requiring Pointer environment to work)
PRINTER UTILITIES DISK 1 (includes label creators and printer tutorials)
PROGRAMMING DISK 1 (includes QL PROLOG)
PROGRAMMING DISKS 2 - 8 (C68 Compiler)
RALF BIEDERMANN DISKS 1 + 2 (excellent collection of programs, games, utilities, etc.)
RECREATIONAL MATHS DISK (includes life cellular automatons etc)
SUPERBASIC UTILITIES DISK 1 (includes large numbers of utility progs, toolkits, procedures etc)
TEXT DISKS 1-7 (The Bible - Old Testament)
TEXTDISK 15-16 (The Bible - New Testament)
TEXT DISK 14 (400+ Business Letters from U.S.A.)

Please note the above are only part of our P.D. Library. New programs are constantly being added and more are always welcome. Please send an S.A.E. for full catalogue.
We can also supply commercial software for STs and Amigas!
ALL PRICES INCLUDE MEDIA AND POST AND PACKING FOR THE UK.
PLEASE ADD 10% FOR ORDERS IN EUROPE. 20% OUTSIDE OF EUROPE
PLEASE INDICATE 3.5"/5.25"/MDV VERSIONS WHEN ORDERING

Perfection

BLOOD, TOIL, TEARS & SWEAT FROM:

DIGITAL PRECISION LTD
222 The Avenue,
Chingford,
London,
E4 9SE

NOW VERSION 3!

EVEN BETTER

FASTER

SMOOTHER

SMARTER

WITH EVEN MORE FEATURES!

AND STILL JUST £79.95*, OR

JUST £119.95* WITH SPELLCHECKER.

EASIER TO USE THAN EVER BEFORE...

SPEED ON STANDARD QL (4-5 TIMES FASTER ON GOLD CARD)

NAVIGATION: INSTANT.

SEARCHING: ABOUT 100K (i.e. 25 PAGES) PER SECOND

BLOCK HANDLING: ABOUT 20 PAGES PER SECOND ETC!!

* DP WISHES YOU A MERRY XMAS! FROM JAN 1ST 1992, PERFECTION WILL COST £89.95 & PERFECTION PLUS £129.95

LIGHTNING SPECIAL EDITION
LIGHTNING

Until the autumn of 1989 the fastest way of speeding up your QL display was to buy **Lightning**, which greatly accelerated QL text printing, graphics and maths, without affecting compatibility at all. NOW you can buy **Lightning** Special Edition, which is significantly faster than **Lightning** and does a lot more! **Lightning** Special Edition is simplicity itself to use. Once it is loaded ALL programs will AUTOMATICALLY benefit from the enhancements it provides. If you are using a QL without **Lightning** you are probably a little pale (quote from John Norton of Sector Software), you should get out and about more... Go to some QL shows or meetings where you will see **Lightning** in action - or take our word for it. If you don't have **Lightning** you are WRONG.
Lightning Special Edition works by automatically (I know we keep using the word, but it is the only one that is really correct here) and instantly replacing QL ROM code (or Minerva code, for that matter - Minerva and **Lightning** complement each other superbly) that has usually been optimised for space, with extremely high speed routines written by us that do the same job but much faster. Screen output speed gets accelerated by factors from over 1.5x to over 10x (about 2x-4x is representative), graphics are drawn twice as fast (points are plotted 5 times faster) and internal maths is speeded up by 2x-5x (you can even vary the precision). There is virtually no cost in RAM (for example, you can still run Quill with a fairly large document on an unexpanded QL with **Lightning** Special Edition).
The Special Edition is supplied on EPROM plus disk/cartridge: if you already have something precious plugged into the QL's EPROM socket (at the rear), there is no problem - all the EPROM's functionality is duplicated on the other medium!
Lightning Special Edition provides more than acceleration - you can dynamically adjust channel parameters - like ink, paper, font, screen position, use over 80 fonts, a null device, a character drain and all sorts of other interesting gadgets.
Lightning Special Edition installation has been totally automated, and will not present you with complications no matter how computer-naive you are.
If you cannot afford the Special Edition, get **Lightning**. Refer to its review in September 1988 QL World to see how effectively **Lightning** acquitted itself.
Both of these programs transform the QL into an altogether more zippy, business-like, efficient, enjoyable machine.

PC CONQUEROR WITH DR-DOS v5.0
PC CONQUEROR

Terrific though we know the QL to be, we do feel the pressure to be "PC compatible" in today's world. There is increasing demand to be able to bring home and run the programs we use at work (or the other way around!), and to have access to the vast storehouse of PC software: word processors, databases, spreadsheets, expert systems, accounts and financial modelling packages, vertical market applications, visualisation aids, graphics/CAD/PCB designers, languages/compilers, operating systems, environments, utilities, adventures - you name it, there are scores of each type readily available for the PC. And thousands of shareware/PD programs too, most for the cost of a blank disk plus postage.
If you buy PC **Conqueror**, you will be able to run these programs! To boot up PC **Conqueror** takes 10 seconds from the F1/F2 prompt: thereafter, your QL is a HIGHLY compatible PC clone (indeed, more compatible than some "real" PCs). **Conqueror** is all-software. There is no comparison in quality between **Conqueror** and its predecessor: **Conqueror** has ALL the features of Solution (read the details later in this ad if you are unfamiliar with Solution's legion facilities), but is almost TWICE as fast: this has come about by our careful rewriting and optimising of Solution's code. As if the colossal speedup was not "enough", **Conqueror** (unlike Solution) runs perfectly even with PC software that makes various "non-legal" calls to the PC operating system. **Conqueror** runs with virtually anything that will run on a PC: QL Worlds from December 1989 to March 1990 listed several hundred PC programs/utilities found to work with **Conqueror**. It is simpler to say that we have yet to find a program that runs fine on a standard PC that doesn't run with **Conqueror**: we are aware, however, of programs that will run with **Conqueror** but won't run on some PCs!
Because in **Conqueror** we've cracked the problem of detecting when the PC screen has been changed, we need not slavishly update the screen many times a second (taking precious time away from the main PC-emulation job) as did Solution. Instead, we update the screen instantly it needs to be updated. This simple to understand but very hard to implement modification gives **Conqueror** additional (over and above what we've already discussed), "tunable" acceleration, as well as absolutely smooth echoing of keyboard input to screen (Solution could be a bit jerky when you typed quickly).
Conqueror's new features include a more flexible configurator and a better diagnostic and supervisor option, an enlarged manual (**Conqueror** itself is more compact!) with a troubleshooting chart, and a new mode of operation (in addition to the "normal" one of reading/writing PC disks directly) which allows you to create mini PC environments - you select the size, location and name - on any QL device (including floppy, ramdisk, hard disk and even mdv) which look like files from QDOS (and can therefore be copied with SuperBASIC's COPY!) but are indistinguishable from PC drives from within DOS (**Conqueror** works with all versions of DOS).
If you do not have legal access to a copy of DOS, you need to buy DOS too (DR-DOS or MS-DOS) - but we sell the complete DR-DOS (with Viewmax, Shell, Cache and all system utilities) at c 1/2 price!
Of course QLs are better than PCs - but QLs that are PCs as well are better still. We will leave the last word to people who have already bought **Conqueror**. All these sentiments are unsolicited.
*"I wish to congratulate you on the excellent work you have done on **Conqueror**. The improvements in performance over Solution are astounding. Well done!" B.C.Papegaaij, Netherlands. "I am highly delighted with this new emulator. (Apart from the speed-up) it also appears to be more tolerant." L.Chandler, Peterborough. "Congratulations on bringing such a fast PC emulator into the world - on it, even Wordperfect runs at a reasonable speed." R.Williams, London. "I'm impressed with the improvement in speed over Solution." P.Vervoort, Netherlands. "Thank you for your prompt service. I have **Conqueror** up and running, and congratulate you on an excellent piece of software." G.Leagas, Hartlepool. "On some benchmarks almost as fast as a PC." P.Johnson, Stoke on Trent. "**Conqueror** is still a whole lot faster (even) without Lightning than Solution is with the assistance of Lightning." P.Christie, Glasgow, who went on to praise **Conqueror** for running software Solution couldn't handle. "**Conqueror**, to which I upgraded from Solution, is a delight to use by comparison!" B.Gouldwell, Dunipace..... V.Pakanen, Finland sums it all up rather well with - simply - "Excellent."*

PROFESSIONAL PUBLISHER

To show you a little of what our **Professional Publisher** can do, we have prepared our last advertisement using it. Notice from our May 1990 advertisement how we can wrap the result around graphics or in fact anything, of any shape. When we wrote **Professional Publisher** (PP), we knew it was a very special sort of program. PP can produce pages of quality - virtually indistinguishable from those prepared on professional typesetting kit, the only limiting factor might be your printer: however, while the very best output output from PP will be obtained from 24 pin models and lasers, you will be stunned by what PP can squeeze out of the humblest 9-pin machine. Great care was taken in the design of PP so we were absolutely sure that no actual knowledge of, or practice with, desktop publishers was required in order to use it (the '**Professional**' in '**Professional Publisher**' refers to the output quality, not the level of operating skill required. When you use PP, you will notice that at each and every stage a menu is available (there are getting on for a hundred menus in total) with a list of options selected by using either the cursor keys and SPACE bar, or by pressing a digit key - use what suits you!
There is context sensitive, on-screen help too. When you get more experienced with the program, you may select Command mode (using the Enter key) and choose operations directly, bypassing the menu system. PP is more user-friendly than any page-making program we have ever seen on any computer, period.
Let us talk you through how you might choose to produce a page or succession of pages. This is just one way you might proceed: PP does not impose any sequence of steps upon you, and you can omit certain operations altogether. You will have pre-configured PP to boot up with a generous lot of fonts (you select which ones you are likely to want - of course you can load in additional ones, or discard existing ones, at run-time too). You could then set the required page dimensions and orientation, as well as not-necessarily-symmetric margin, grid, gutter, column and navigation-guide positions (yes, half the PP manual is a glossary) - you could have pre-configured PP for these too, or loaded in alternative layouts (layouts are distinct from page contents) you've created in previous sessions. If you don't set layout we'll use the default, or the one used for the previous page. Now you would plan the page in detail. Laying out graphics (if any) comes next - you can create these in PP itself, with its superb rubber-banding, dozens of brushes, palettes, texture-fills and so on. Alternatively, you can load in screens created elsewhere, including Eye-Q, Easel, any other graphics programs or digitiser, into a cut and paste buffer where a dozen tricks (including resizing, slanting, scrolling and texturing) are available, and then take the finished product onto the page. This done, you might insert headlines or captions, selecting from the dozens of fonts available. Each font can be manipulated in billions of ways (yes, we mean thousands of millions): to give but two examples, you have a choice of 32 slopes for italics for the font, and dozens of aspect ratios are selectable. Now you might opt to get the main body or bodies of text down on the page. As fonts are defined to great accuracy (upto 2304 pixels PER CHARACTER!) jaggedness is a thing of the past, and visually the choice of fonts can only be described as stunning! You can do this either by directly typing it into cursor-dragged boxes (with all the options you would expect from a dtp system, and a few more besides), or by loading it in from file created by Quill, **PERFECTION**, Editor or other word-processor. The latter method is better (because you retain the text as a character stream rather than as pixels when you save the file). Highlights such as bold, underline etc which you may have inserted into the text are preserved. Indeed, you can control PP's operation from within the text file itself. If you are an advanced user, you can even teach PP your own mnemonics, so that it switches between different styles and modes as it encounters instructions you put into your text file when you created it! The imported text file is editable within PP. It is up to you to decide where the text is to lie - PP places no restrictions on either the number or the shape of the windows into which the text is to flow: they need not be rectangular, and can have any irregular border, and can even overlap or be contained one inside another! You can freehand-draw (there's excellent rubber-banding to help you) the window borders as you choose, to get any effect you desire, to fill any space you wish and to avoid any existing material already on the page (or to reserve room for new material). Amazingly - within the window the text will all be perfectly micro-justified in the font(s) of your choice, however bent or contorted you made the border. Text will flow automatically from one window to the next either until you have run out of text or out of windows. There are many text formatting facilities: you can select word-wrapped, force-broken or hyphenated, and you can specify minimum numbers of "pre-hyphen" and "post-hyphen" characters so that absurd hyphenations are avoided (if no sensible hyphenation position can be found the word is wrapped instead). There are so many fine-tuning controls here that the rest of this ad could be devoted to describing them and would still leave things out! We will have to content ourselves with but one example: with micro-justification (pixel by pixel spacing, not crude character by character stuff) we even allow you to specify what % of padding space is to be allocated between characters and how much between words! Text work completed, you can then put in the final touches by adding borders, shadows, patterns or designs, overwriting or slipping under or combining these with existing material, repositioning parts of the page if necessary. The end result - be it for a letter, letterhead, document, manual, article, newsletter, magazine, book, thesis, ad - is far better than you have any right to expect from a piece of software costing under £2,500, let alone under £100...

PROFESSIONAL PUBLISHER TOOLBOX

NEW!

For Professional Publisher users - this useful addition not only supplies several man years worth of beautiful high definition fonts - including familiar types like Roman and Universal - but also contains many smaller fonts, more clipart and programs to load Sector Software clipart, filter text before importing into Professional Publisher, save parts of Professional Publisher pages as screens (for importing into any graphic program - like Eye-Q - or manipulating via SuperBASIC) etc. Excellent value.

FONT ENLARGER

For Professional Publisher users - loads of large fonts are automatically created by this multitasking utility, as and when you need them (or in advance), by enlarging existing smaller smaller fonts from PP itself and from Lightning Special Edition and hordes of other sources: with this there is NO jaggedness at all. A font editor for small and large (hdf) fonts is included.

GRAFIX

Scaleable output for all our desktop publishers on 9- and 24- pin printers: a useful alternative to the built-in drivers.

EYE-Q

There is no way to describe **Eye-Q** except as the best graphics program for the QL. This master is now four years old, and we have never felt the need to change anything. Its use is characterised by absolute simplicity, speed and power - it has that indefinable precision "feel" that is just right. All the expected manipulations are provided. Whether your needs are technical drawing, labelling, design, illustration, freehand work, copying - or just having fun, **Eye-Q** will not disappoint. Of course it is menu driven with context-sensitive help. The system takes 5 minutes to learn. The variable zoom and fill facilities, anti-fingerslip measures, cursor acceleration and so on make **Eye-Q** a classic in its own time.

ULTRAPRINT

To get the best printer output from Eye-Q or any other graphics program from any other source, **Ultraprint** delivers. An amazing 22 styles to choose from: enhance contrast (for line output) or gradation (for pictures) and vary magnification... A printer without **Ultraprint** is no printer at all.

MEDIA MANAGER SPECIAL EDITION
MEDIA MANAGER

MMSE is a joy to use. Whether something has gone wrong with a disk or tape ("Not found", "Not a valid Quill file", "Bad or changed medium", "Read/write failed" etc) or whether you want better control over your programs and data, **MMSE** should be to hand. Virtually any calamity can be recovered from automatically: all permutations (accidental deletion or part-overwriting, part-formatting, errors yielding: bad map but OK directory, bad

directory but OK map, bad map and directory, OK map and directory but bad file sectors, unknown fault, power glitch corruption and so on) have been carefully thought through and catered for. If nothing is wrong, but you just want to explore and understand more about your system, you can potter to your heart's content, assisted by the clear and packed-with-facts manual. Dozens of different diagnostic printouts can be produced. The whole system is menu-driven, with context-sensitive, on-screen help for every option. The speedy Sector Editor is a positive delight: the collector file facilities, bulk recovery, auto-navigation, skipping through the medium in physical, file (if map), logical (if no map) or uncollected/logical (if destroyed map, and because of "chequered" history with lots of overwriting/deletions no one-step recovery available) sequences must all be experienced to be believed. **MMSE** is extremely simple to operate, and assumes no advance knowledge whatsoever.
Alternatively, if you wish to tidy up your disks or cartridges, **MMSE** allows you to change volume format names, sort directories into alphabetic, date or size order, analyse file contents and histories, change case of filenames, move data/programs to/from alien-format disks, introduce or break copy-protection systems (illegal use prohibited!), **MMSE** can and will deliver the goods. It is absolutely superb.
The standard **Media Manager** is much less powerful, and less easy to use. It is only for those on a tight budget.

TOOLKIT III WITH ROM
TOOLKIT III

Virtually everyone with a disk system has Tony Tebby's fine TK2 Supertoolkit on board (usually built into the disk interface). **Toolkit III** - which works whether or not you have TK2 - takes off where TK2 ended, adding about 70 new commands and enhancing many existing QL and TK2 commands. **TK3** is for everyone with a QL. You can get this system on cartridge/disk, with or without a plug in ROM cartridge in addition. The documentation is complete and very comprehensive. Some of the added commands are:
ADIM% * ADIMN * AND_L * ATYP * BASREF * BV_BASE * CHANNELS * CH_BASE * CINT * CLOSE% * DEVLINK * DIR_USE * DIT$ * DIV_L * EOR_L * EXTRAS * FACC * FLP_SEC * FLP_START * FLP_TRACK * FLP_USE * FRAC * ISFLT * ISINT * KEY$ * LARRAY * LOWER$ * MEMCOPY * MEMSWAP * MJOB * MJOB_W * MOD_L * NFS_USE * ODD * OM_INIT * ONPIPE * OR_L * PEEK_F * PEEK$ * PEND * PIPE_ * POKE * POKE_F * PRED * QDOS$ * QIN * QOUT * QTEST * QWAIT * RAM_USE * REPLACE * REPLACE$ * RESET * RJOB_A * ROUND * SARRAY * SEARCH * SETDIR * SETDIR_A * SETHOST * SETNET * SETRO * SETRW * SETSYS * SETUSER * SGN * SORT * SORT_I * SUCC * TK3_EXT * UPPER$ * USER * WN_BASE * WSETHOST * WSETNET * WSETRO * WSETRW * WSETSYS * WSETUSER
Whether or not you can program, **Toolkit III** is of great use!

QFLICK CARD INDEX SYSTEM

Few users actually require all the facilities of a complicated database like Archive. **QFlick** presents a very convenient alternative - a very fast, simple to use card-file database, with easy to learn, snappy search and navigate commands and clean file-handling. You can move Archive data to/from **QFlick**. You can run multiple copies of **QFlick**. And **QFlick**'s data is organised so it is easy to program from SuperBASIC, even for tyros!

PERFECT POINTER TOOLS

This excellent program gives you an on-screen pointer (arrow) environment of incredible smoothness, and 6 utilities with it. To explore the world of QPtr, Things, Hotkeys, Window Manager....

QKICK MULTITASKING SYSTEM

A pull-down menu controlled multi-tasking front-end, ideal for running in the background and giving you notepads, file-handlers, quick backup, clock, diary, mini-database, calculator etc etc.

DISKTOOL WITH QUICKDISK

An exciting way to accelerate disk access by upto 30%, add password protection to disks and to optionally increase disk storage capacity by 36K to 1512 sectors! All this works while maintaining full compatibility and normal disk control...

DIGITAL C SPECIAL EDITION
DIGITAL C COMPILER

Superb C compilers these - fast in execution, they produce extremely speedy and concise code. No-nonsense documentation is included. The Special Edition has many more features, including pointers, long pointers, structures, >64K code sizes, direct access to traps and vectored utilities, and is twice as fast because of its more efficient C/QDOS libraries.

TURBO BASIC COMPILER + TOOLKIT

This state of the art system will automatically convert ordinary SuperBASIC programs - the sort you buy, write yourself or type-in from magazines - into machine code, the language of the 68008 CPU, the brain of the QL. Such pure machine code programs run "directly", without the need to be interpreted by any intermediary system. This direct execution makes them MUCH faster in execution than BASIC. **Turbo** also adds a host of useful high-speed commands (called "toolkit extensions" if you are fond of jargon). Here are some timings all carried out on a JS Trumpcard QL, to give you a taste of just how much **Turbo** can improve things:

	Iterations	SuperBASIC	Turbo'd	Speedup
Empty FOR...END FOR Loop	30000	49 sec	1.3 sec	38x
Empty REPEAT Integer Loop	30000	151 sec	2.4 sec	63x
String concatenation	3000	448 sec	0.4 sec	110x
Search through memory	300000	1410 sec	1.5 sec	900x

Turbo's automatic conversion process, called compilation, is as simple as this: (1) Boot up with the **Turbo** disk (2) Load in or type in your BASIC program (3) Enter the word CHARGE, and watch the friendly front-end menu pop into view (4) Choose a filename for the machine code task that is to be generated and (5) Press the SPACE bar. **Turbo** does the rest! Compilation is a one-off process, and is very fast too - it takes little more time than LOADing the original program did! Once compilation is finished, you have a machine code version of the original program. Start this with EXEC, just as you used to invoke the original program with LRUN: besides the tremendous difference in running speed, you will notice that the program loading time is cut down to a few seconds at most (big SuperBASIC programs can take half an hour or more to load). The EXEC mechanism also allows you to multitask programs, something impossible with SuperBASIC, as well as manipulate their time-priorities, link them together, exchange data and even share parts of their code while executing.
If you are an advanced user, **Turbo**'s numerous fine-tuning facilities, 200-command toolkit (a terrific complement to the famous Supertoolkit) and 300+ page manual will be irresistable. If you are a beginner, you will wonder how you ever did without **Turbo**'s program diagnoses and auto-correction.
Turbo is more than a very clever optimising compiler. **Turbo** is magic. If you do not have it, you can have no conception of the experience you are missing and the power you are forfeiting.

SOLUTION WITH DR-DOS v5.0
SOLUTION

This program transforms your QL into a pretty compatible - albeit not fast - PC clone. **Solution** will run over 95% of the "big name" PC software you have read about, missing out only on programs that make illegal use of the PC's operating system. **Solution** works solely from software so you don't have to worry about ripping your QL to pieces to fit anything, or have anything hanging out of the back. Just boot up the **Solution** disk and you will be using a PC, which will then ask for a copy of DOS (any) (just as it would if you were using a "real" PC). End of story - you are now using a PC. There are very few restrictions: both mono and colour CGA graphics are supported. 479K is available for PC software on a 640K machine and 667K when using Trumpcard - more than you will get on a PC or XT! Speed can be increased by using Lightning Special Edition but in final analysis just can't compare with Conqueror's speed). Because your newly aquired PC is really a QL you can multitask two or three PC programs (try doing that on a "real" PC!). You can also run QL programs alongside PC programs (DON'T try that on a "real" PC!). Converting files (data in either direction) between QL and DOS is no problem and you can re-configure the QL keyboard if you wish.

PROFESSIONAL ASTROLOGER
PROFESSIONAL ASTRONOMER

Our use of the term "Professional" in the name of an application program does mean that the quality achieved will meet or surpass the highest professional standards for that application. The term does NOT mean that you have to have the knowledge of a professional in order to get the best out of the programs.
Astrologer teaches you astrology from scratch, and enables you to produce reams (if you are short of paper, you can choose exactly how much) of narrative printout giving a person's horoscope, personality delineation, year-to-year life overview, detailed day-to-day (in fact, minute-to-minute!) predictions, as well as two-person compatibility interpretations. Also provides all the technical readouts, charts and zodiacal wheels you would expect. It is extraordinarily fast (there is a great deal of very clever maths within it) and it performs the whole computation in under a second. The author of the manual is the author of this advert, so you can expect a lucid and humorous read! Whether or not you believe in astrology - indeed, especially if you do not - this program is one that you cannot afford not to have. Scores of detailed readouts for famous people are supplied, incidentally - very interesting reading they make too... Discover Mrs Thatcher's secret yearnings, explore yourself, play the Stock Exchange...
Astronomer is an extremely efficient solar system computer, with planetarium views, planet faces (with shadows/eclipses), five different co-ordinate systems, 1sec=1day cinerama, etc.
Astrologer + Astronomer is supplied at a very low combined price.

ACT SPECIAL EDITION

The **Adventure Creation Tool** is for every programmer or putative programmer. Whether or not you have any interest in adventures, you will find something useful here. Animated graphics, data compression, language design and parsing, maps, object-oriented control and much more, with an excellent educational manual too.

3-D PRECISION CAD SYSTEM

2-D and 3-D design and manipulation, at a speed sufficient to permit real-time animation! Whether or not your interest is serious, **3DP** will change the way you look at the world around us. The variation of viewpoint, perspective and magnification is very smoothIn addition to dot-matrix output, plotters are catered for.

SUCCESS

Run CP/M programs on your QL! What more is there to say, other than that after the PC family, no more common system exists than CP/M, with thousand of cheap programs... And **Success** is fast!

THE EDITOR SPECIAL EDITION
THE EDITOR

If your needs are for a technical Editor, or for full access to the entire ASCII character set (to handle machine code or compressed data files), or if your budget cannot stretch to **PERFECTION**, then this is the program for you. Editor is command-line driven and programmable. The Special Edition version is certainly better than the standard version: that is because the standard one contains only as many features as we could get to fit into an unexpanded QL. Both are fast and flexible, and very powerful indeed in the hands of the intelligent. Not a word processor, Editor's a way of life.

SPECIAL DESKTOP PUBLISHER
DESKTOP PUBLISHER

Both these WYSIWYG ("What You See Is What You Get" dtp systems are excellent in their own rights - it is only when you compare them with the stunning Professional Publisher that you become aware of their shortcomings. You won't get fonts as large or smooth as with PP, or wrap-around graphics, or as sophisticated a printer driver or text/graphics file import facility. You will get a very workmanlike tool, capable of producing output that the computer press described as fantastic and superb.... The standard edition is the ideal if you do not have a disk drive: if you do have one, go for the Special version, which correspondingly has more features including textures, large windows, better drawing and improved command entry. All upgrades are possible, and there is only a £10 penalty for doing it in two stages. So if you simply cannot afford PP, one of this pair is certainly for you.

SUPERFORTH COMPILER WITH REVERSI

Why not learn FORTH, the most logical computer language of all? This superb FORTH-83 compiler produces stand-alone multi-tasking code of speed comparable to C. **SUPERFORTH** source is even portable to other machines! The manual teaches you the language.

IDIS SPECIAL EDITION
IDIS

Machine code (from other people's programs, toolkits and the ROM) is unintelligible until you put it through IDIS, the intelligent disassembler. IDIS Special Edition automates everything it possibly can, and requires no human intervention. It even sorts out subroutines, replaces addresses with names, untangles data from code and so on. Standard IDIS contains as much as we could pack into an unexpanded machine, and is nearly as automatic.
If you want to find out how computers work, buy one of these two!

CPORT BASIC TO C PROGRAM TRANSLATION SYSTEM v1.16
This program translates SuperBASIC programs directly into C source code, automatically! If you want to move programs into C for migration to other hardware, or want to get your programs running faster, or simply want to learn C the easy way (chuck BASIC in one end and examine the C that spews out the other), CPORT is the system for you. CPORT is extremely friendly, easy-to-use and tolerant of poorly-written BASIC. There is even a method of dealing with BASIC toolkits. The C it will generate is very readable, human-like and is often optimal. Of course, the better the quality of the BASIC you put in, the better the clarity of the C that will be generated. But don't misunderstand - even if your BASIC is a rats-nest of GOTOs and GOSUBs, CPORT won't mind. Usually, the generated C - which can even be switched between the ANSI and Lattice (K&Rish) industry standards - needs no tinkering with. The only conditions worth mentioning are that there must be no computed branching (e.g. GOTO 3*Y+X), no interpreter-only commands (LIST, EDIT, RENUM etc) and that if the program contains PROCs or FNs (as it probably should!), there mustn't be any GOSUBs *as well* - not restrictions at all. CPORT is an amazing program, making breakthroughs in AI. CPORT is available on its own or together with the acclaimed C68K compiler.

SUPERBASIC MONITOR
Yes - this program monitors and reports on the performance of SuperBASIC programs as they run (i.e. dynamically) under the interpreter. Even if you only occasionally tinker, this one you must have! Ideal with XREF (static analyser), BETTER BASIC, TURBO etc.

XREF v3.10
An incredibly competent program analyser - structure, the dynamic call hierarchy of procedures/functions, step-ladder report, glossary, warnings, variable usage and so on.

QMON v2.05
The ultimate version of Tony Tebby's superb machine-code Monitor. An absolute must for those who really want to know what's going on in the QL. £10 off if you return the old Digital Precision Monitor.

COMPARE v1.07
This little gem compares files (data or program) at great speed, and allows shuffles and alignment in auto- and semi-auto mode. You cannot do without it!

MEGA DICTIONARY
If you have at least 1.5Mb RAM (Goldcard, some Thors and STQLs) this is the ultimate PERFECTION PLUS accessory, enabling the best possible spellchecking. It contains well in excess of 300,000 words: about 50% larger than the largest of the dictionaries supplied with SPELLCHECKER! Specify if you have HD or ED disks, for then we can supply this massive dictionary as one file (else as 2). Another attraction of this masterpiece - it will complement accessories (wordgames/crosswords/anagrams/wordfinds etc) that we intend to soon release. This is an incredible product, far better than you imagine!

SPELLCHECKER
SPELLCHECKER works with and without PERFECTION. SPELLCHECKER can always spellcheck files - either PERFECTION format or plain ASCII (e.g. Editor, Quill or text87 export files, for example). If PERFECTION is present, SPELLCHECKER can also selectively (pages/blocks/all) spellcheck the current document, or spell as you type. Spellchecking speed is over 35,000 characters per second on Goldcard, and a cracking 3 pages per second on lesser QLs. SPELLCHECKER comes with three ready-made dictionaries: small, medium and large. If you have only a 256K expansion, you can only use 'small'; if you have only 640K, you can use 'medium' too; if you have Trumpcard 768K (or more) you can use any of them. You also get a utility for creating and maintaining (add/delete/edit/view) "user" (i.e. your own) dictionaries. Further, you can spellcheck using either one or two (typically a supplied one and a user one) dictionaries - you can even specify which is to be used first when checking! So if you bought PERFECTION without SPELLCHECKER and want to add it (PERFECTION PLUS = PERFECTION + SPELLCHECKER), or even if you don't have PERFECTION at all, this is the product for you!

QMATHS MATHEMATICAL SYSTEM
An incredible mathematical compendium for the QL. Pride of place goes to the symbolic problem solver.... It can solve problems, simplify expressions, factorise, expand etc etc - all symbolically! If you could sneak this one into a maths examination (school/GCSE/O/A/S/undergraduate) you would have a formidable ally. It knows about all the algebraic operators, powers, roots, brackets (any number), trigonometry, matrices, determinants, vectors, factorials, perms and combs, binomials, exponentials, logarithms, hyperbolics, inverse functions, infinite series and their approximations, complex and imaginary numbers, conversions, and even calculus - both differential and integral (even knows definite integration, integration by parts etc)! And when the program is working something out, you can opt to get it to display some or all of the steps either All this is accompanied by a superb interactive tutorial. So whether you have been terrified of maths or are a boffin, this is the program for you: no mathematical skills are assumed or needed. Whether all you want to do is compute 2+2 or d/dx((sin(x)+x.log(x))^(x^(g(x)))), QMATHS will do it. There is nothing like this available on any computer. In addition to this program, the package also contains an interpretive, fractal image-generating language with loads of beautiful fractal programs supplied for you to use, modify or adapt. No programming skill is assumed or needed. In addition, there is a multiple precision floating point maths package - allowing calculations with all the QL functions at precisions up to over 600 decimal digit accuracy (that is not a misprint) and very fast too. In addition, there is a 3D surface modelling program and lots of calculating routines to perform all sorts of algebraic and statistical computations in your own BASIC or Abacus systems. This is an incredible package.

PROFESSIONAL PUBLISHER TOOLBOX PART TWO
Did you think we'd stopped? Another blisteringly good collection of fonts and utilities for the Professional Publisher user, augmenting and adding to the first toolbox. You really should have both.

HARDBACK v1.4
THE hard disk utility - full/incremental backups, searches etc.

RECOVER v1.11
Recovers lost Archive databases. Ideal with Media Manager SE.

ARCHDEV + RTM
The Archive development environment - gives enhanced speed, greater workspace and a cleaner boot system.

DATABASE ANALYSER
Fast statistics about your Archive database(s).

ARCHIVE TUTORIAL
Everything you always wanted to know about Archive - but were afraid to ask. Interactive and competent.

NAMES + ADDRESSES
MAILMERGE v2.1
DAT-APPOINT
Names and addresses, mailmerging and appointment diaries - sophisticated, fast and ready to run under Archive. If you've never used Archive, now is the time to start!

SEDIT v1.10
SCREENPRINT
Creates/edits and prints screen-format files in Archive.

PRDIT
A fast, sophisticated printer driver manager for the Psion programs - replaces the free one!

CASH TRADER v3.2 INCLUDING ANALYSER v3.3
PAYROLL
An accounts system for the small and medium-sized business, for the layman as well as the sophisticated user, with lots of excellent reporting and management facilities.

MICROBRIDGE
Bids and plays for the defence at Contract Bridge. Includes a superb bidding tutor. Manual a masterpiece. Understands and obeys ACOL and many other conventions.

SUPERCHARGE SPECIAL EDITION
If you have an unexpanded QL, or cannot afford TURBO, but want SuperBASIC programs to go faster, Supercharge is the answer. It has about half the speed of its big brother, is not so tolerant of badly-written programs, and lacks many of TURBO's features (like linking, program sizes over 64K, 200 command toolkit etc).

SUPER SPRITE GENERATOR
SSG moves things about the screen rapidly, at machine code speed, directly from simple SuperBASIC. Any number of sprites (each with up to 16 frames for smooth and realistic motion), 256 speeds, 256 planes, collision detection and dozens of special effects.

SUPER ASTROLOGER
A cut-down version of the Professional Astrologer - smaller horoscopes and manual, no interpretations for forecasting or compatibility testing. A marvellous buy at the price!

BETTER BASIC EXPERT SYSTEM
SuperBASIC is a super BASIC. If you want to improve your programs automatically, and learn as you do this, get Better Basic.

TRANSFER UTILITY
Copies files between devices, performing translates as it goes. Can move your microdrive material onto disk, so programs run from disk but you still have access to microdrives.

	£		
PC CONQUEROR WITH DR-DOS v5.0	169.95	eT	
THE SOLUTION WITH DR-DOS v5.0	119.95	eT	
PERFECTION PLUS WITH SPELL CHECKER	119.95	dT	
CPORT WITH C68K COMPILER	99.95	cT	
CASH TRADER v3.0 + ANALYSER	99.95	cT	
PC CONQUEROR	89.95	eT	
PROFESSIONAL PUBLISHER	89.95	cT	
CPORT BASIC TO C TRANSLATION SYSTEM	89.95	cT	
PERFECTION	79.95	dT	
TURBO BASIC COMPILER + TOOLKIT	79.95	aT	
PROFESSIONAL ASTROLOGER WITH ASTRONOMER	69.95	aT	
QMATHS MATHEMATICAL SYSTEM	59.95	cT	
PROFESSIONAL ASTROLOGER	59.95	aT	
LIGHTNING SPECIAL EDITION	49.95	+aT	
PROFESSIONAL PUBLISHER TOOLBOXES (PAIR)	49.95	eT	
ACT SPECIAL EDITION	49.95	eT	
THE EDITOR SPECIAL EDITION	49.95	dT	
SPELLCHECKER	49.95	dT	
MEDIA MANAGER SPECIAL EDITION	49.95	dT	
3-D PRECISION CAD SYSTEM	49.95	dT	
PAYROLL	49.95	cT	
HARDBACK	49.95	cT	
SUCCESS	49.95	bT	
DIGITAL C SPECIAL EDITION	49.95	aT	
TOOLKIT III WITH ROM	49.95	+a	
THE SOLUTION	39.95	eT	
IDIS SPECIAL EDITION	39.95	dT	
SPECIAL DESKTOP PUBLISHER	39.95	ct	
QMON MACHINE CODE MONITOR v2.05	39.95	at	
SUPERFORTH COMPILER WITH REVERSI	39.95	aT	
EYE-Q	39.95	aT	
MICROBRIDGE	39.95	d	
MEGA DICTIONARY	29.95	gT	
PROFESSIONAL ASTRONOMER	29.95	fT	
PROFESSIONAL PUBLISHER TOOLBOX	29.95	eT	
PROFESSIONAL PUBLISHER TOOLBOX PART TWO	29.95	eT	
TOOLKIT III	29.95	a	
SUPER SPRITE GENERATOR	29.95	a	
SUPERCHARGE SPECIAL EDITION	29.95	a	
QFLICK CARD INDEX SYSTEM	29.95	a	
PERFECT POINTER TOOLS	29.95	a	
GAMES COMPENDIUM (ALL FIVE GAMES)	29.95	a	
THE EDITOR	29.95	at	
DIGITAL C COMPILER	29.95	at	
XREF	29.95	aT	
SEDIT	29.95	aT	
ARCHDEV + RUN-TIME MODULE	29.95	aT	
SUPER ASTROLOGER	24.95	f	
DESKTOP PUBLISHER	24.95	d	
QKICK MULTITASKING SYSTEM	24.95	a	
MEDIA MANAGER	24.95	a	
IDIS INTELLIGENT DISASSEMBLER	24.95	at	
SUPERBASIC MONITOR	24.95	aT	
LIGHTNING	24.95	aT	
BETTER BASIC EXPERT SYSTEM	24.95	aT	
FONT ENLARGER	19.95	eT	
NAMES AND ADDRESSES	19.95	dT	
DAT-APPOINT	19.95	dT	
DISKTOOL WITH QUICKDISK	19.95	b	
ULTRAPRINT	19.95	aT	
SCREENPRINT	19.95	aT	
RECOVER	19.95	aT	
PEDIT	19.95	aT	
MAILMERGE	19.95	aT	
DATABASE ANALYSER	19.95	aT	
COMPARE	19.95	aT	
ARCHIVE TUTORIAL	19.95	aT	
GRAFIX	14.95	eT	
SUPER BACKGAMMON GAME	9.95	f	
REVERSI GAME	9.95	f	
DROIDZONE GAME	9.95	f	
BLOCKLANDS GAME	9.95	f	
ARCADIA GAME	9.95	f	
TRANSFER UTILITY	9.95	b	

** KEY:*
+ As well as cartridge or disk, you get a ROM
a Available either on cartridge or disk
b Available only on disk
c Minimum 512K exp:only available on disk
d Minimum 256K exp:either cartridge or disk
e Minimum 256K exp:only available on disk
f Available only on cartridge
g Minimum 1.5Mb RAM:only available on disk
t Compatible with THORs including the THOR XVI
Compatible with all THORs except the THOR XVI

SPECIAL DEALS: 5% off total if you buy 2 programs/upgrades
10% off total if you buy 3 programs/upgrades
15% off total if you buy 4 programs/upgrades
20% off total if you buy 5 programs/upgrades
25% off total if you buy 6+programs/upgrades

For TERMS AND CONDITIONS see that section in any of our last 12 advertisements in the pages of QL World.

********** ORDER FORM **********

Send to: DIGITAL PRECISION LTD, 222 THE AVENUE, LONDON E4 9SE or phone 081-527 5493

Please send me: _____

Name: _____ Address: _____

Postcode: _____ Signed: _____

Encl: CHEQUE/VISA/ACCESS/MASTERCARD/EUROCARD/PO/MO/CASH for £ _____

Card No: _____ Expiry: _____

DIGITAL PRECISION LTD

Delete as appropriate: (QL/THOR/STQL) (MDV/3.5"/5.25") (Disk=360/720/1440/2880/6400 sectors; nb 2sectors=1K) (*If* 720 sectors:SSDD/DSSD)

NOTICE BOARD

Many QL users feel isolated. This is one of the messages we get, sometimes in letters and sometimes when we talk to dealers. They don't actually write to us to say "I feel isolated!", but when a problem crops up they may find that they have nobody to turn to who has the answer, and this is where *QL World*, *Quanta*, and the QL dealers and publishers come in. They can turn to us, looking for the answer to their problem.

So far, so good, but none of us have all the answers at our fingertips. Dealers in the QL world have a good reputation for helping people who are using one of their programs or hardware extensions, even if they are using them in novel ways. OK, let's face it, occasionally you come across an impossibility, but most of the time there's an answer out there. The challenge can be: finding it.

Even the people who can find the answers, in fact, especially the people who can find answers, often have their hands so full that finding time to investigate a puzzle is a real challenge. It's like juggling an assortment of fascinating objects. If you drop one, it breaks; if you drop a different one, it bounces, and yet another different one rolls quietly away into a corner and lies there until you can pick it up again.

Of course, in times like these, everybody in the business is juggling with things which are all urgent – developing this hardware, completing that software, checking these orders, mailing those items, getting all the little bits of QL World to arrive in the right place at the right time (and there are plenty of times when this doesn't work) – in other words, with a high proportion of 'breakables'.

Then there is the fact that many QL users are working on projects which are very personal to them, especially unusual bits of programming. We are all working with the same tools, with a common language of *SuperBasic*, *Quill*, *Abacus* and *Archive*, *Toolkit 2* and *Turbo*, the Trump Card, the 8056 printer – but so many variations on the themes are also in use, and some quite weird and wonderful ones. (Weird? Wierd? Another exception to the rules. They used to call it 'wyrd' when I was young, back in the 9th century. Things do change, but they don't get any simpler!) The QL itself has different rom versions, and all the upgrades and extensions that have been produced since 1984 have their own rules.

There is nothing like experience for getting to the heart of a difficulty. This is why I have appealed to readers with experience to offer advice to people who are stuck.

You won't be surprised to hear that we have had a good response. More are always welcome, of course, and ad hoc questions will also appear. So if you have a QL-related query, drop us a line – keep it as short and as simple as possible, because – if it is puzzling you, well, it'll surely puzzle someone else – cross your fingers, and wait.

Let's see if we can break the isolation down.

Z 88

EaZy Money	£71.00	Centronics Interface	£22.25
Z-Base	£71.00	128K Eprom Pak	£28.00
Z-Tape	£51.50	128K RAM Pak	£38.75
Z-Term	£51.50	Z88 Computer	£223.00
ThinkZ	£51.50	512K Ram Pack	£103.50
Superchip 4	£154.00	256K Eprom Pack	£68.25
Golf	£25.50	1Mgb Ram Pack	£220.00
Spellmaster	£60.50	Squeeze 88	£47.00
Topper	£10.50	Superchip 3	£133.50
		Z88 QL Link	£27.50

PSION SERIES 3

128k Computer	£200.00	256k Computer	£250.00
128k Flash card	£40.00	256k Flash card	£70.00
512k Flash card	£120.00	1MgB Flash card	£200.00
2MgB Flash card	£300.00	128k Ram card	£80.00
512k Ram card	£200.00	1MgB Ram card	£300.00
Parallel interface	£30.00	Mains adaptor	£15.00
PC link	£70.00	Mac link	£70.00

TK COMPUTERWARE

the QL stockist

Tel: 0303-81-2801

PSION ORGANISER 2

Model LZ	£100.00
Model LZ 64	£130.00
Comms Link	£60.00
Comms Link – Mac	£73.00
Travel Pak	£40.00
Harvester Superchip	£100.00
Letter Organiser	£40.00
Informed Organiser	£40.00
Diary Link	£13.00
128k Datapak	£100.00
64k Datapak	£60.00
32k Rampak	£60.00

SPARES

Keyboard Membrane	£8.50
JS ROM set	£31.00
ZX 8301	£15.50
ULA ZX8302	£10.50
MAB 8049	£7.50
QL Power Supply	£31.00
MC 68008	£26.50
M/drive assembly-(complete)	28.00
Microdrive Assembly (exchange)	£21.50
Voltage regulator (high output)	£5.00
QPower Regulator	£25.50
QL Test kit	£14.50
Repairs	phone

SUNDRIES

VDU cleaning kit	£7.50
Disk Cleaning Kit 3.5"	£15.50
10 Microdrive cartridges	£30.00
100 Microdrive labels	£4.00
Discs 3.5" 10 DSDD (branded)	£26.00
Power Clean Mains Filter	£24.50
Joystick + adaptor	£18.50
Psion 2.35 Set	£23.50
Printer Ribbons	Phone
Keyboard-90 Interface	£77.00
Tandata QCOM set	£90.00
QL Scanner	£143.50
QUIMI I/F with mouse	£101.50

SOFTWARE

BUSINESS

Project Planner	£30 00
Sales Invoice	£20.50
Small Traders Pack	£30.00
Stock Accounting System	£41.00

TEXT

Master Spy	£31.00
Page Designer 2	£36 00
Perfection	£80.00
Perfection +	£125.00
Professional Publisher	£90.00
QTYP 2	£31.00
SPY	£15.50
Turbo Quill Plus	£14.00
The Editor	£30.00
The Editor Special Edition	£50.00

FINANCE

Home Finance [Buzz]	£22.50
Investment Monitor	20.50

CAD

Eye Q	£40.00
GraphicQL Plus	£25.50
PCB 1	£102.50
PCB 2	£51.50
TechniQL	£51.50
Technikit	£25.50
The Painter	£51 50
3-D Designer	£39.00
3D Precision	£50.00

PROGRAMMING

Assembly Workbench	£25.50
Basic ally	£20.50
Better Basic	£25.00
Digital C	£30.00
Digital C Special Edition	£50.00
Forth	£31.00
IDIS	£25.00
IDIS Special Edition	£40.00
Macro Assembler - GST	£39.00
Macro Assembler - Talent +	£20 50
Pro Fortran - 77	£101.50
Pro Pascal	£101.50
QMON 2	£20.50
Lightning Special Edition	£60.00
Locksmith	£15.50
Mailfile	£15.50
Media Manager	£25.00
Media Manager Special Edn	£50.00
QFlash Ramdisc + t'kit	£19.50
QFlash Ram Eprom	£25.50
QFlash Ram Utilites	£10.50
QPAC 1	£23.50
QPAC 2	£51.50
Q Top	£31.00
RPM	£15.50
Sidewinder Deluxe	£20.50
Super Toolkit 2 (Eprom)	£30.00
Super Toolkit 2 - Reconfigurable	£30.00
Success	£40.00
Terminal	£31.00
Touch Typist	£12 50
Utraprint	£20.00
XRef 2.00	£20.50
4 Matter	£15.50

LEISURE

Bridge Player 2	£15.50
Cosmos+	£15 50
Fight Simulator	£19.50
Matchpoint	£15.50
Micro Bridge	£40.00
Prof Astrologer + Astronomer	£80.00
Prof. Astronomer	£40.00
Snooker	£13.50
Strip Poker	£15.50
Super Astrologer	£25.00
Super Backgammon 3	£13.50
Type 22	£18.50

GAMES

Aquanaught 471	£19.50
Arcanoid	£15.50
BJ in 3D and	£12.50
Brain Smasher	£18.50
Crazy Painter	£13.50
Crown of Sorkinen	£18.00
Deathstrike	£15.50
Firebirds	£15.50
Hopper	£13.50
Hoverzone	£15.50
Jungle Eddy	£15.50
Karate	£15.50
King	£13 50
Land of Havoc	£19.50
Lost Pharoan	£15.50
Mortville Manor	£15.50
Stone Raider	£13.50
Super Arcadia	£10.50
Vroom	£13.00

UTILITIES

Archivist	£31.00
Archivist MP	£41 00
ArcnRTM+ArcnDEV	£20 50
Cartridge Doctor	£18.50
Conqueror	£100 00
Conqueror MS-DOS 4.01	£125.00
Conqueror with DR-DOS 5.00	£180.00
Discover	£31.00
Lightening	£30.00
Super Forth + Reversi	£40.00
Super Sprite Generator	£31.00
Supercharge Special Edition	£25.00
Turbo	£80.00

• Please telephone for details of other products not listed here •

BOOKS

Assembly Language Prog	£9.00	Using Your Z88	£10.00
Z88 Dabhand Guide	£16.00	Z88 Computing	£10.00
Z88 Ref Manual	£16.00	Z88 Software DLV Guide	£40.00
Z88 Magic	£16.00	Using Psion Organiser	£10.00
Z88 Pipedream	£16.00	Psion Handling Book	£15.00

ALL PRICES INCLUDE VAT. U.K. POSTAGE £2.50 PER ORDER. - WE EXPORT WORLDWIDE

Phone for details of other QL products OR send SAE for our latest catalogue.

Credit card phone orders accepted, or send cheques, postal orders, Eurocheques, to:

TK COMPUTERWARE, STONE STREET, NORTH STANFORD, ASHFORD, KENT TN25 6DF

TELEX 966676 PMFAB G

VISA · Access · MasterCard · AMERICAN EXPRESS · EUROCARD

FAX 0303 812892

QL SCENE

New pocket machine from Psion

Psion UK have come up with a new pocket computer – the Series 3. The diminutive, 'clam-shaped' computers, approximately 6 x 3 x .75 inches in size, not only improve on the well-known Psion Organiser range, but advance and extend the technology and capabilities of pocket computing.

Coming in two versions, a 128K at £199.95 and a 256K at £249.95, the computer arrives complete with seven applications programs including a database, a word processor, an alarm, diary and time planner, a calculator, a telephone tone dialler and the programming language OPL. It can multitask programs and has its own window graphics operating system, using the built-in 240 by 80 pixel lcd screen. It also has a communications port giving it the capability of exchanging data with desktop computers and printers, and can be expanded to 4 megabytes using the optional plug-in solid-state disks.

We understand that it will communicate with the QL and we expect to be publishing a review soon.

For more details of the Series 3, contact **Psion UK, Alexander House, 85 Frampton St., London NW8 8NQ. Tel. 071 258 7368.**

Ecoutez

QL Contact France have contacted QL World to say that they are still alive and well. Get in touch with them at **QL Contact France, 38-40 Rue Stephenson, 75018 Paris, France.**

CARE OFFER

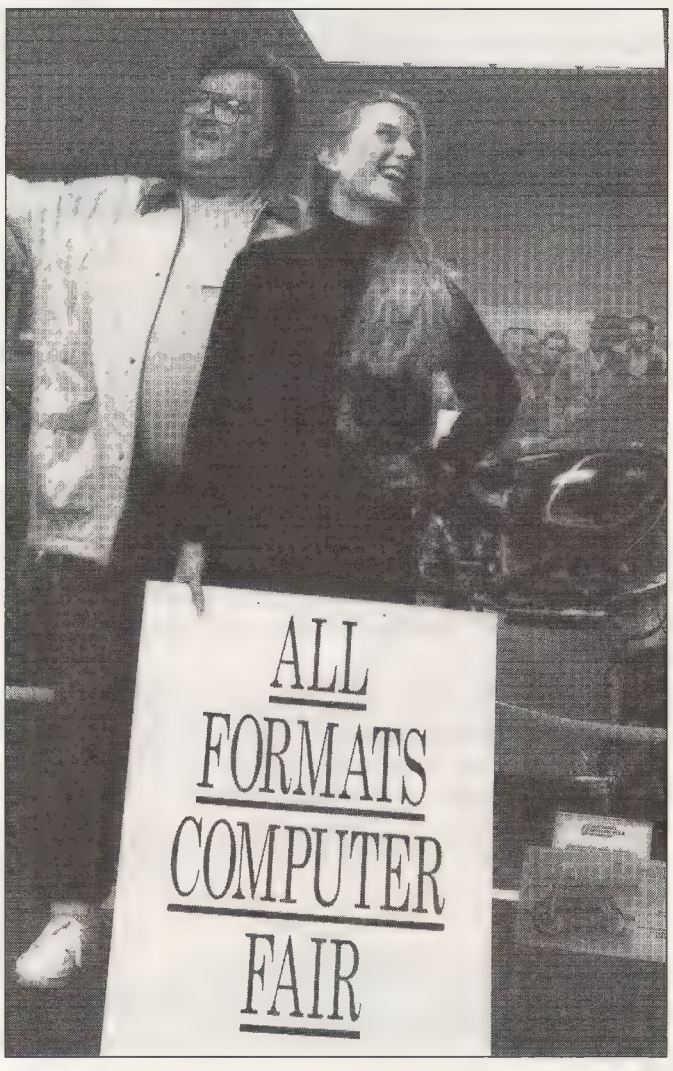
Care Electronics are doing a special offer on Qpac 2 during January and February 1992. From the beginning of January to the end of February you can save £10 on the normal price of Qpac2 by paying £39.48 plus £2.35 post and packing. Something to spend your Christmas money on! **Contact Care at 800 St. Albans Road, Garston, Watford, Herts WN2 6NL. Tel. 0923 894064 for orders and information.**

Family Format

What's this? It seems that the All Formats Computer Fairs have got themselves a lovely blonde to assist them with their publicity. But they are keeping it in the family – the lady is graduate technology designer Christian Riding, daughter of long-standing Fair agent John Riding, he of the advance tickets and stand sales.

"Christian is hard-working, dynamic and outgoing, qualities that will help ensure the continued success of the Fairs," says owner and organiser Bruce Everiss – qualities which also characterise the dedicated programmers, suppliers, publishers and contributors of the QL Community, some of whom are nearly as attractive as Ms Riding, too, in the right light!

Dates for forthcoming fairs are: Motorcycle Museum, opposite the NEC, M46 junction 6, 11 January; University Sports Centre, Leeds, 12 January; Horticultural Hall, Westminster, London (the fair most likely to attract QL support), 18 January; Brunel Centre, Templemeads, Bristol, 19 January; City Hall, Candleriggs, Glasgow, 26 January. Fairs run from 10am to 4pm and admission is £4. Advance ticket sales and information from **John and Christian Riding on 0225 868100.**



OPEN CHANNEL

Open Channel is where you have the opportunity to voice your opinions in *Sinclair QL World*. Whether you want to ask for help with a technical problem, provide somebody with the answer, or just sound off about something which bothers you, write to: Open Channel, Sinclair QL World, 116/120 Goswell Road, London EC1V 7QD.

Versatile

I am writing to you first to congratulate you on keeping *QL World* as interesting as ever and secondly to ask if it is possible to include even more information for the novice.

Although I have had a QL since they first came out, I am still having difficulty with finding out how to do things.

As we all know, the manual is not a lot of help, due to many misprints and wrong information.

I must say that nearly all the things I have learned have come from the QL User/QL World magazines.

In my case, I use *Archive* extensively, and one of the best things that ever happened to me was the publication in October, November, December 1985 of the QL User *Owner's Manual* series, and the inclusion in part three of the Stamp Collectors' program. I do not collect stamps – but I have been able to adapt this program to cover twenty-seven other items, including such things as books, house contents, record collections, etc.

Although the current *New User Guide* is a good idea, and very helpful, it does not, as yet, go quite so far as including programs that can be adapted as above.

A R Kempton
Polygon
Southampton

Editor's Notebook

I have mixed feelings about Christmas. On the one hand, there's red holly berries and rich green leaves, frosty air, carol singing, shiny wrapping paper, log fires in the pub, crackers and port. On the other hand, there's fighting in the kitchen, tears at bedtime, somebody's bound to be sick after the Jelly course, and that's just the grownups!

I have similar feelings about December's *QL World*. Beautiful cover, good games, useful utilities, international reports, and the biggest collection of production grollies in a month of Sundays. In true Blue Peter style, we even included half a page That We Had Prepared Earlier.

Well, these things happen. All the working parts seem to be in order. I'll tell you what: an extra Christmas Present to the person of unusual taste and sensitivity who spots the boob that nobody noticed (and nobody minds) except me.

And a Happy Christmas and New Year to everyone.

Back-out

Everyone knows that backing up all your disks on a regular basis is the only way to guarantee that all your effort is not in vain. Wrong! I keep three copies of my *Archive* programs, and the data. One day I got the bad medium message on the work disk. After reformatting the disk and copying a backup onto it, the same message came a week later at a different place when the disk was loading. After this happened about four times, I jumped to all kinds of conclusions – dirty or misaligned heads, maybe, or even a disintegrating circuit.

I formatted the problem disk on my Archimedes, which displayed a list of bad sectors as it formats and automatically reformats until as many sectors as possible can be salvaged, then maps the rest out. After verifying that all the sectors were good, the disk was set aside for about a week and was verified. A number of bad sectors showed up again. That disk was retired to the circular file on the floor.

The second time that happened with a program disk, the reason dawned on me (I hope). They were the first disks I bought after acquiring the disk drives in 1987. The magnetic material on the disk surface could not retain its magnetism for so long any more.

Like microcassettes, disks wear out in time. It just takes longer; I wonder if that is also true for hard disks as well?

J Paul Bissonette
Otterfing
Germany

Editor's comment: Are these 3.5 in disks or 5.25 in disks? That information would be of interest. I have heard of cassette tapes made in the 1970s which began to show significant deterioration within five years, regardless of the amount of use received, whereas tapes made more recently, when tape technology had advanced somewhat, still functioned well after many more years. The 3.5 in disk units are more modern, inherently more rigid and better protected from outside influence than 5.25 in disks. The same is true for hard disks, but moreso.

It's unwise to assume that any medium is indestructible – even the Rosetta Stone was cracked when they found it. But in general, the longer the technology has been tried and tested, the more reliable it will be.

AND/OR

Referring to K Dunbar's Mandelbrot program in the September *QL World*, the problem is in line 120. This line should check the absolute value (abs) of a plus b - then it will work. In the listing given, his OR code is only checking three

Mandelbrot set plotter

```
10 OPEN #4, con_512x2356a0x0
20 PAPER #4, 0:CLS £ 4
30 SCALE #4, 256,0,0
40 real=-2,1:image=-1,25
50 FOR horizontal=1 TO 256
60 FOR vertical=1 TO 256
70 x=real: y=imag
80 a=x: b=y
90 count=0
100 REPeat colourselec_loop
110 n=(x^2-y^2)+a
120 m=(2*(x*y))+b
130 IF n>2 THEN EXIT colourselec_loop
140 IF n<-2 THEN EXIT colourselec_loop
150 IF m<2 THEN EXIT colourselec_loop
160 IF m>-2 THEN EXIT colourselec_loop
170 count=count+1
180 IF count>=100 THEN EXIT colourselec  loop
190 x=n:y=m
200 END REPeat colourselec_loop
210 IF count>=100 THEN INK #4,0
220 IF count>=75 AND count<100 THEN INK #4,2,0
230 IF count>=50 AND count<75 THEN INK #4,2
240 IF count>=25 AND count<50 THEN INK #4,7,2
250 IF count>=10 AND count<25 THEN INK #4,7
260 IF count>=5 AND count<10 THEN INK #4,4,2
270 IF count>=2 AND count<5 THEN INK #4,4
280 IF count<2 THEN INK #4,0,4
290 POINT £4, horizontal, vertical
300 imag=imag+9, 7656E-3
310 IF vertical=256 THEN real=real+6, 8421E-3
320 IF vertical=256 THEN imag=-1,25
330 AT 0,0 PRINT real: AT 0,15:PRINT imag
340 END FOR vertical
350 END FOR horizontal
```

out of four quandrants and the AND code is only checking one out of four. There are other ways of approaching this, but this would seem to be the simplest.

Simon Goodwin
Warley
West Midlands

Chess check

I'm happy to inform you that I have received my disk box. It arrived some days after I sent my letter to you. Thanks a lot.

Does anyone have any information on *Psion Chess* – ratings, how many positions it analyses per second, results in tournaments it has been playing, etc? Personally I have version 2.01, but I am lacking some of this information. I have the Atari version of the program.

Oyvind Vir
Dorheim
Norway

Editor's comment: this is one for Emulator users.

Manual help

I recently had my 8056 printer manual stolen from my car. Can anybody supply me with a copy of the manual? I will pay for copying costs. Please contact me via QL World.

A Landaw
London NW9

Mandelbrot

Those readers who enjoy mathematical recreations might like to try out the enclosed program which plots the Mandelbrot set. Despite the limited palette of colours available in high resolution mode on the QL, this program gives a quite strikingly attractive rendering of the set. The set itself is in black; it is the boundary areas which are coloured.

A warning – on my Thor 1 the complete plotting took some 24 hours! So those who have compilers would be well advised to use them.

Parts of the set can be 'zoomed' by changing the parameters on the real and imaginary axes.

The set is fully described by A K Dewdney in four articles in Scientific American in the issues for August 1985, November 1987, February 1989 (including the basic algorithm) and June 1989. These should be available as back issues from Scientific American, or for reference through some public libraries.

Experimenters can try juggling with the colours in lines 210 to 280.

Values for the set run from -2 to .5 m real (x) axis and from 1.25 to 1.25 on imaginary (y) axis. To zoom on portions of the set, lines 40, 300, 310 and 320 are to be modified by entering the new starting points for the real and imaginary on line 40 and then setting the increments in lines 300 and 310 by dividing the range by the horizontal and vertical steps (in this case, 380 and 256).

Line 330 is not strictly necessary – I put it in to show you where you are for determining interesting boundary areas of the set for later 'zooming'.

Frank Gutteridge
Corsier
Switzerland

Postscript

And last but not least, back to Capt. Starling's letter on the Perfection Manual last month . . .

Deep in the heart of Mr Churchill's own constituency, Freddy Vachha selects a suitable quotation from his wide acquaintance with the great man's sayings. "This is something up with which I will not put," he says firmly. "Capt. Starling asks why we don't put the important bits at the front. How much front do we have? The reference to justification and the differences between *Perfection* and *Quill* are in Section 1, subsection 1.1, on the first page of the manual!"

And they are, too. It isn't really fair to say that the instructions are hidden in a page called 'line setting' – centring and justification *is* line-setting. A decade ago nobody would have talked about Justification when they meant Formatting, but the innocent process of Justification (described with great accuracy by Freddy in his paragraph on pseudo-spaces at the top of page 27) has taken a trouncing in recent linguistic migrations and is now generally used to describe any sort of line or paragraph formatting – O tempora, o mores! It isn't only computer terminology which confuses people.

You can indeed use Perfection without the aid of a manual, but as with any word processor you must tread gently in places. The HELP screen has Justification near the top, and everything else on one 'page'. Even so, starting life with a new word processor by attempting to Centre a headline is jumping in at the deep end – if you get into trouble it can be difficult to intuit your way out. In fact, Eric Starling's problem wasn't with the Justification commands, but with the Reformatting – something which Quill, unlike the vast majority of word processors, does automatically as you type.

Most word processors don't do this because it causes drastic reductions in speed. Instead, they offer a choice of formatting lines and paragraphs to order, reformatting entire documents before printing to enforce the formatting commands, and the better ones (including Perfection) also allow the entire document to be configured in advance to follow a particular formatting convention. You don't have to worry about this normally because the defaults are set up to allow typing and printing of justified text. This doesn't include centred headlines, though, because not many people use them.

The section on menus is introduced in section 2.1, and the section which describes the individual commands is roughly between pages 45 and 83. The trouble with putting the important things at the top is that everyone has different priorities, and there is never enough top to go round.

My favourite quotation (not, I think, from Sir Winston) is: "When all else fails, read the instructions," but the prophet should have added: "For anything more complicated than a can-opener, allow about an hour."

A Question of Dots

Geoff Wicks scans the desktop publishing field.

Desktop publishing for the QL has made great strides in a space of a few years from the simplicity of the first *Front Page* to the sophistication of *Professional Publisher*. With the increasing complexity of the programs the demands on memory have increased. Front Page could be used on an unexpanded QL, was about 30 K long and and could only fit a half page (40 K) in the available memory. Professional Publisher is about 200 K long and has a maximum page size of almost 375 K. To be able to use this page size an 896 K *Trump Card* is necessary and no more than one page can be saved on a 3.5 in disk. I cannot resist the temptation to add that less than 10 years ago we were all struggling to fit everything into the 1 K of a ZX81!

For me there is simple test of the quality of a desktop publishing system. The Workers' Council of which I was formerly secretary is legally required to produce an annual report, which must be sent to every employee and to various official agencies. The report is about 7000 words long and became a prestige document in which we invested a great deal of effort. It was produced traditionally with typewriters or wordprocessors, cut and paste and photocopying. The only recent innovation was the use of computer fonts for headlining. I had used QL desktop publishing programs for announcements of council elections and meetings, but convincing my council colleagues that the annual report could also be produced this way was a different matter.

The Full Scope

At one time I would not even have made the attempt, partly on technical and partly on aesthetic grounds. I have experience of the *Front Page* series, *Desk Top Publisher Special Edition* and *Professional Publisher,* but have never used either *text 87* or *Page Designer*. Of the programs I have used, only *Professional Publisher* has the technical scope to allow the easy importing of large quantities of justified text to be printed in an attractive and easily readable font. This was necessary before I could convince my layman colleagues that a better report could be produced using desktop publishing.

One of the connections between the technical possibilities of a desktop publishing program and its aesthetics is the possible resolution. Commercial typesetting systems have a minimum resolution of 600 dots per inch (dpi). A laser printer has a resolution of 300 dpi. An A4 page in *Desk Top Publisher* or *Professional Publisher* is 960 pixels wide — a resolution of 120 dpi and 800 pixels long — a resolution of 72 dpi. An A4 page produced on the *Front Page* series is 800 pixels wide, a resolution of 100 dpi. Put another way, there is a limit to the quality you can produce unless you have far more memory than is available to the average QL user. The software houses producing the programs have to work within this restriction.

This also explains why the original *Front Page* was the desktop publishing equivalent of the ZX81. It deserves a place in history as the first attempt to provide desktop publishing on the QL, and was the starting point for more sophisticated programs. It taught the fundamentals of desktop publishing but was not suitable for serious work. All text had to be input manually in windows that did not cover either the full width or full length of a screen, let alone a page, and the only fonts that could be used were low resolution QL ones.

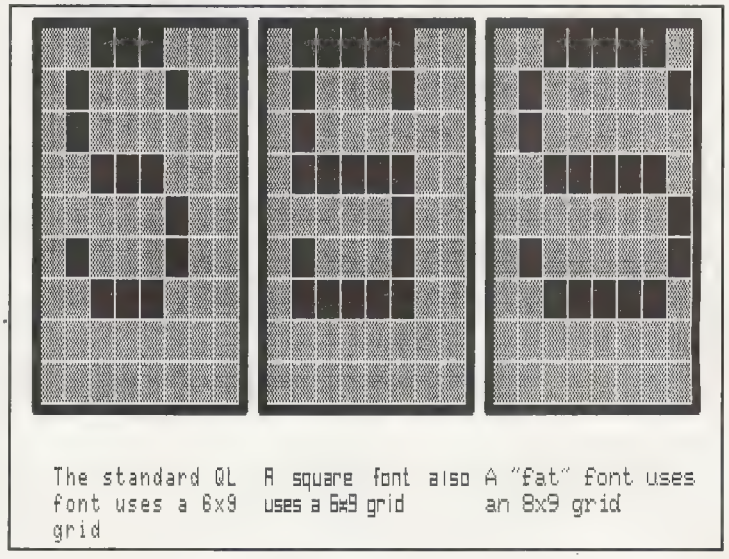
QL fonts are produced on a matrix eight pixels wide and nine pixels high, although in practice since CSIZE 0,0 is only six pixels wide two of the columns are not used. Effectively the matrix is 6x9. The larger CSIZES are magnifications of this grid, so that in practice the maximum resolution possible for character widths 0, 1, 2, and 3, when used in desktop publishing, is equivalent to 120 dpi, 120 dpi, 80 dpi and 60 dpi. The grid is too small for certain letters such as m and the w to be reproduced legibly in the smaller CSIZEs. If the larger CSIZES are used the familiar steps in the letters can be seen. Aesthetically it is not pleasing.

In the early days of desktop publishing there were a number of attempts to alleviate this. The step effect is most noticeable in curved letters such as O or diagonals such as Z. 'Square' fonts were sometimes used to keep the number of curves and diagonals to a minimum, but although they could be used for small quantities of text such as announcements, they did not look natural in larger documents. Another possibility was the fat fonts in which the full 8x9 grid was used to design the characters instead of an 6x9 grid. This gave a slightly better resolution than the standard characters, useful when designing an m or a w, but the disadvantage was that only CSIZES 1,0 and 1,1 could be used. Digital Precision supply several attractive versions of fat fonts with their desktop publishing programs and with *Lightning.* Unfortunately many of the fonts are too fancy for text use in a serious document.

The only other possibility of improving the quality of documents using a standard QL font is at the printing stage. Later versions of the Front Page series allowed printing with more than one pass of the printing head. A better quality printout was obtained by a very slight displacement of the printing position for the second pass. Professional Publisher has a more sophisticated printer driver which offers an 'Interpolation' option in which 'a suitably computed pixel row is inserted between any two 'regular' rows. If you possess Professional Publisher the effect can be seen by printing out a small amount of text produced using the standard QL font in three different ways. First — with one printer pass, second — with two passes and third — with one pass but selecting the interpolate option.

Improvements

Front Page Extra was a great improvement on Front Page. Not only could justified text be imported in columns from Quill doc files, but also high resolution fonts using grids larger than 8x9 were introduced. The text importing routines were fairly simple. The user only had the choice of the number of columns per page and not of the width of the columns. The possibilities of highlighting text with bold or italic characters were very limited, since the user had to stop inputting to change the



The standard QL font uses a 6x9 grid

A square font also uses a 6x9 grid

A "fat" font uses an 8x9 grid

font. Only complete lines could be highlighted and only standard QL fonts could be used. Nevertheless within these limitations the importing of text files worked faultlessly.

The high resolution fonts were few in number but were useful for headlines. The width of the characters had to be a fixed number of bytes. The maximum width was 8 bytes and in theory the fonts could be of any height. A grid of 64x64 was thus possible. No proper font editor was provided so that if I wished to change a character or devise a new font, I had to fall back on a rather complicated procedure of saving the font as a screen, modifying this with *Eye-Q*, and then reloading it into Front Page. A further disadvantage was that each character had to be of the same width. With the smaller fonts it was necessary to choose between a width of one byte and have unclear m and w, or two bytes and have a large white gap on either side of i. The larger the font size, the more acute this problem of fixed character width became.

At about the same time, Digital Precision released *Desk Top Publisher* and shortly afterwards *Desk Top Publisher Special Edition*. At first sight these seemed to offer more facilities than Front Page. The range of high resolution fonts was much greater and more professional in appearance, they could be manipulated in various ways to produce italic, bold and inverse versions and a variable character width was possible. A good font editor was supplied as part of the program. Unfortunately these fonts also had their disadvantages. They were all produced on a 16x16 matrix, although a special technique for defining the base line position of lower case characters with descenders (g, j, p, q and y) made this equivalent to a 16x24 matrix. The only way to obtain a larger character size was by the magnification of a smaller font, with the inevitable appearance of jagged and stepped edges. Another shortcoming was that the full Ascii keyboard set (characters 32 to 127) was not supported.

Importing

The text importing routine was also full of promise, since not only could the QL character set be used, but also high resolution fonts. Windows into which text was to be inputted could be opened anywhere on the page and highlighting took place automatically. I found however one snag with the importing routine. I could never get it working satisfactorily unless I resorted to inputting paragraph by paragraph.

In this period I found myself longing for the ideal program combining the best of Front Page with the best of Desk Top Publishing. The next best thing was to use a combination of both when designing a page. I even devised means of converting the high resolution fonts from one program to the other. When it came to text inputting, I opted for a low resolution QL font and Front Page. I even went so far as to make a mock-up of a few pages of my annual report using one of Digital Precision's fat fonts and Front Page's input routine, but the result was disappointing and unlikely to convince the layman that desktop publishing techniques gave a better result than traditional ones.

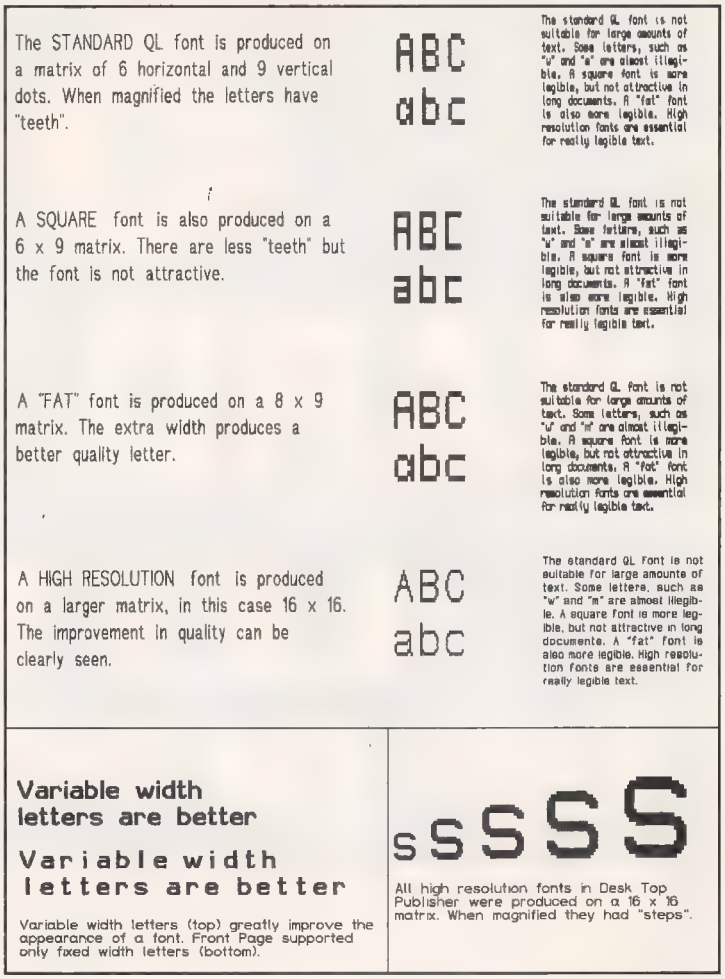
When my ideal program finally arrived in the form of Professional Publisher it surpassed all my expectations. The high resolution fonts really were high resolution and could be on grid sizes from 8x8 to 48x48. The full Ascii range (characters 32 to 191) is supported. Furthermore not only did the text importing routines work almost perfectly with both QL and high resolution fonts, but the text could be made to flow from window to window, saving valuable time. Occasionally I have had problems with text emphasis which usually occur when two emphasis changes are made at the same time (for example, printing bold underlined text). However when this has occurred a slight change to the text to be input has usually been sufficient to set the matter right.

One of my small criticisms of Professional Publisher, although I suspect that not everyone would agree with me, is that there are too few good high resolution fonts provided with the program suitable for use with large amounts of text, a situation which has been partially remedied by the new fonts in *Professional Publisher Toolkit*. Having said this I would however warn against having too high an expectation of the results of using fonts. In practice the maximum font size for use with large quantities of text could not be greater than a 12x12 matrix, which is better than the standard QL font, but no better than the resolution of the NLQ mode of a dot matrix printer.

The fact is that the quality of desktop publishing on the QL given traditional hardware constraints is of fairly low resolution where only *standard* width and length printing is available, 120 dpi compared to the 300 dpi of a laser printer and the 600 dpi of typesetting systems. The only way that resolution can be improved is by using some form of reduction at the reproduction stage. If you want better quality you have to produce a page on A3 or A4 and reduce it at the printing or photocopying stage to A4 or A5. You have then boosted your resolution to 180 dpi, and this technique is used by many small publishers using traditional as well as dtp methods.

Just imagine, though, the quality you could achieve if you could design a page the size of a broadsheet newspaper and reduce it to A4. Professional Publisher can do just that! The printer driver provided supports fingerprinting, in other words, printing at a quarter of the size of the page. It provides a horizontal resolution of 240 dpi and a vertical resolution of 144 dpi. The quality has to be seen to be believed. For me it is the printer driver that places Professional Publisher in a class of its own, and makes it the only QL desktop publishing program for the very serious user.

You can guess the rest. I produced a mock-up of my annual report and was pleased with the results. Only one thing stood in the way of convincing my colleagues that we must go over to desktop publishing — unfortunately Professional Publisher had come too late, and I was no longer a member of the Workers' Council!



QL HARD.WARE

LEVEL 2 up grade chip

For those who have early QL memory expansions, Rich Mellor tests a cheap way of upgrading.

INFORMATION:

Product: FLP/RAM Level 2 replacement chip for the Trump Card or SuperQBoard

Price: £15 (plus £4 post and packing)

Supplier: Jochen Merz, Im stillen Winkel 12, W-4100 Duisburg 11, Germany.

For those of us who cannot afford the upgrade to a *Gold Card*, but wish to make the most of their existing equipment and get extra speed at a low cost, this new chip can be a great help.

I am the proud owner of quite an early *Trump Card* (although Miracle have been kind enough to upgrade the rom on the board for me in the past) and having decided that I did not (as yet) need the extra memory and power provided by the Gold Card, this little wonder seemed the ideal half-way house.

What you actually get is a small micro-chip in a protective holder and a new manual explaining the new features provided by it. I was disappointed to see that Jochen did not provide any instructions on how to fit the chip (contrary to his normal helpfulness), but this did not cause me any difficulties in practice, although some users may be rather disconcerted by this.

Simple fit

After switching off the QL and removing the Trump Card, fitting the chip was in fact quite simple (I do not know if it would be more difficult on the SuperQBoard for which there is a similar chip available); I merely had to lift the cover off the Trump Card (I just gently prised off one side — there was no real need to undo the fiddly screws which hold down the other side of the cover) and identify the chip to replace. It was not very difficult to spot, because (on the old Trump Card) there were only two chips of a similar size. The chip which has to be replaced is the first chip on the left-hand side of the card (looking from the edge connector) which lies parallel with the left side. You need to use a flat bladed screwdriver and a lot of patience to prise the chip out of the socket, noting which end of the chip has the notch in it (this should be the end nearest the edge connecter). Once this has been removed, simply push the new chip home using gentle pressure and being careful not to bend any of the legs, plug the Trump Card into the QL and switch on the power.

If your QL powers up correctly the F1/F2 screen should display free memory of 896K (depending on whether you have the full Trump Card), Trump Card 2.09 and Toolkit 22.23 (or higher). If this is not shown, then check that the Trump Card is correctly inserted before assuming you have failed to insert the chip correctly.

Once this has been done, it will not be obvious that the new chip has been inserted until you try to access the disk drives or microdrives. Disk access is noticeably quicker than on the earlier rom, but unfortunately it is difficult to give accurate speed-up ratios for the new rom, because with the new chip installed the QL makes greater use of what are known as 'slave blocks' — areas in memory used to speed disk access by storing parts of files temporarily. This is most obvious on COPY and SAVE operations where your program can continue working well before the end of the SAVE operation. The example below shows the difference that this can make (although the overall saving is minor). It does however mean that programs are improved two-fold.

Loading speed

Programs like the Psion package which only load part of a long file at a time are speeded up somewhat because more of the file is actually held in the slave blocks, therefore the program does not have to access the disk quite so often. On a test file on *Quill*, this speeded up downward

scrolling by 20 per cent, and upward scrolling by 10 per cent.

Control is returned to the user more quickly, allowing you to continue typing in Perfection (for example) much sooner.

Speed of file access is not all that the new chip can offer. With the new chip installed, you now have similar facilities as on the Gold Card for real sub-directories and better file handling.

Directories

The new manual which you will receive with the chip gives details of how to use the new commands to create sub-directories on disks, hard disks or ramdisks (you cannot currently create sub-directories on microdrives, but then you should not have that many files on a microdrive); together with how to set up your expansion board for your disk drives, and there is even a little section giving some information about ramdisks.

The first group of commands in the four page manual is the most important, because these allow you to set up your system to take account of the type of disk drives attached: the command FLP_START and FLP_STEP allow you to set the start-up time for your disks and the step-rate respectively (only the latter command will be new to Trump Card users). The step-rate should be set to the rate suggested by the disk drive manufacturer (if you have any instructions for your disk drives) — I purchased my one-third height drives from Miracle in 1986/7 and have found that the highest step-rate (3) was necessary to prevent the disk drives from groaning every time that they are accessed.

Security level

The old Trump Card command (FLP_SEC) which allowed you to alter the amount of checking out when a file is saved or a disk formatted no longer works with this new chip — instead the highest security level is always chosen automatically.

Other new commands allow you to set/read the update date, the backup date and/or the version number of the given file. The update date is the date and time when the file was last altered (for example, saved or copied to). In contrast, the backup date is generally used to contain the date and time when the file was last copied from (this will however be zero unless it is set using the command SET_FBKDT). I would have liked to have seen amendments to the COPY and WCOPY commands so that the backup date of the file being copied was set to the current system date and the new file retained the same update date as the original (of course the backup date of the new file would have to remain zero as now, because this has not been copied from).

The version number of a file can be useful if you are developing a program, so that you can keep track of which version of a file is which. Unfortunately, each time that you save a file, the version number is reset to one — maybe the SAVE function should be altered so if a file already exists it asks if you (a) want to overwrite the file and (b) update the version number. So far as I have been able to ascertain, unless you use the given command (SET_FVERS) to alter the version number by hand, the only command which will automatically increment the version number is OPEN which updates the version number (unless you tell it otherwise by issuing the command SET_FVERS #channel before closing #channel) when the file is later closed.

Sub directories

The final utility provided by the new chip is the ability to have real sub-directories. These are really only of any use to users who have access to a hard disk and will no doubt override the rather poor implementations of sub-directories which exist on some of the QL hard disk systems (of course they are a big help for the extras high density disk drives, but then you need a Gold Card to access those anyway!).

The original Trump Card rom and *Toolkit II* provided several commands to move up and down a 'directory tree', including DUP, DDOWN, DNEXT, PROG_USE . . . , but these were a little difficult to use and did not help to put the files on a given disk into any sort of order. To complement these commands, the new chip introduces two new commands MAKE_DIR and FMAKE_DIR which both perform the same job, but the latter command is actually a function and enables you to trap any error codes easily.

New commands

The new commands allow you actually to 'place' different files into a separate sub-directory. For instance, your disk may contain the following files:

```
boot
quill
program1_source_bas
program1_manual_doc
program1_source_asm
program2_source_bas
program2_source_asm
```

Entering the command MAKE_DIR f1p1_program1_ will then group all of the files together which begin with the prefix 'program_', altering the display of DIR f1p1_ to:

```
boot
quill
program2_source_bas
program2_source_asm
program1 ->
```

As you can see this is a very useful way of ordering your disks whether they be floppy disks or hard disks. What is even more unusual is that any attempt to delete the file program1 (which forms the sub-directory) gives the error 'in use' if there are any files contained in that sub-directory and any attempt to rename the file program1 gives the error read only (whether there are any files there or not — contrary to the manual!).

Forced default

Entering the command DIR flp1_program1 will only list those files contained in that sub-directory. You can force the QL to accept the given sub-directory as the current default by using the standard Toolkit II command PROG_USE, after which DIR alone will list all files contained in that sub-directory. Unfortunately this is not as helpful as it may seem, because some programs refuse to recognise the Toolkit II default directories (eg FLP_USE f1p1_program1), so that whenever a program tried to access FLP1_, they would in fact be accessing the sub-directory FLP1_program1_.

Putting the failings of some software aside, this little chip is a major improvement for all the old Trump Card and SuperQBoard users, and the price is low enough to entice the less well-off to upgrade their expansion board. The speed of accessing files is noticeably smoother and I can well believe the advertised claims that this device driver is twice as quick as the original. If you cannot afford a Gold Card, then this may well be the next best thing.

For the means of testing the new chip, I created a test document on Quill of 11788 words (35 pages long). The document was then tested on a Minerva QL (1.92) with TurboQuill+ (2.35) and Lightning SE (2.11).

Test run 2 was with the new chip installed.

1) Load document into Quill
2) Scroll down through document
3) Scroll up through document
4) Go from top to page 22

TEST 1	TEST 2
43.10 secs	16.54 secs
6 mins 54 secs	5 mins 35 secs
14 mins 49 secs	3 mins 23 secs
3.44 secs	3.05 secs

The overall average speedup on Quill itself was therefore some 14 per cent.

Similar speedups in relation to loading the document were experienced with regard to Perfection (the document, after converting to Perfection format, loaded some 51 per cent more quickly), although scrolling on Perfection is mainly unaffected due to the way in which Perfection works.

MIRACLE SYSTEMS

MASTERCARD VISA

QL GOLD CARD

£375 inclusive (£330 export)

• 2 year warranty • 14 day money back guarantee •

- The **GOLD CARD** is the ultimate expansion for the QL. It speeds up program execution by a factor of about four, increases the memory to 1920K bytes and enables the QL to interface to 3.2M Byte disk drives (as well as 720K and 1.44M).
- There is a battery backed clock so that the QL powers up with the correct time and date.
- We offer part-exchange for old expansions:

Part Ex. Item	GOLD CARD price	
TRUMP CARD 2	£225.00	(£200.00 export)
TRUMP CARD 768K	£275.00	(£241.00 export)
TRUMP CARD 256K	£300.00	(£263.00 export)
EXPANDERAM	£325.00	(£284.00 export)
DISK CARD	£325.00	(£284.00 export)
RAM + DISK I/FACE	£325.00	(£284.00 export)
Other Expansion	£350.00	(£305.00 export)

- All the functions on the TRUMP CARD, i.e. TKII, printer buffer, screen dump, RAM disk and memory cut, are carried over to the **GOLD CARD**.
- The floppy and RAM disk drivers now support sub-directories.

JOIN QUANTA !

QUANTA is the Independent QL user's club. If you have not already joined then consider the following:

- QUANTA stands for "QL Users and Tinkerers Association" and provides a forum for users to get together.
- Members requiring guidance on software and hardware can seek out expertise from fellow members.
- Local sub-group meetings are regularly held throughout much of Britain.
- There are also workshops where we and others demonstrate our wares. Members have access to a large software library and can often find spare parts and second hand bargains at QUANTA meetings and can see new products.

The £14 (£17 overseas) subscription includes an informative monthly magazine. Most of all, though, it is good fun so contact the Membership Secretary, Bill Newell, 213 Manor Road, Benfleet, Essex, SS7 4JD, or telephone (0268) 754407 now!

Tel: (0904) 423986

To place an order by phone please have your credit card ready. For overseas customers we charge the prices shown in brackets.

To order by post, please fill in the form opposite or write to us quoting your credit card number and expiry date, or enclosing a cheque payable to MIRACLE SYSTEMS Ltd.

To: MIRACLE SYSTEMS Ltd, 25 Broughton Way, Osbaldwick, York, YO1 3BG, U.K. Tel: (0904) 423986

Please send me _____

I enclose a cheque to the value of: £ _____

Or debit credit card ☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐

expires ☐☐/☐☐

Name _____ Signature _____

Address _____

_____

_____

MIRACLE SYSTEMS

MASTERCARD VISA

QL 3.2 Megabyte Dual Disk Drive

New standard for the QL

£175 inclusive (£155 export)

• 2 year warranty • 14 day money back guarantee •

The **3.2M byte DUAL DISK DRIVE** together with the GOLD CARD brings the QL system right up to date. The drive contains 2 mechanisms each able to store 3.2Mb on an ED diskette. ED actually stands for 'Extra High Density' and refers to the rate at which data bits are read from and written to the disk. ED disks transfer data at 1 Megabit per second whereas HD (High Density) transfers at 500K bit/s and the old QL standard DD (Double Density) at 250K bit/s. Please note that these are instantaneous rates and that to work out file transfer times other factors must be considered.

The drive is fully backward compatible in that it can read and write both DD and HD diskettes with any type in either drive. The DD, HD and ED diskettes are all the same physical size known as the 3.5" form factor. The HD and ED diskettes have an extra hole in their casing, in different places, so that the drive can detect which type of diskette has been inserted. This means that there is no need for the user to tell the QL of the diskette type in use. For example, if you format a DD it will format to 720K, whereas an ED would format to 3.2M.

The drive is fully cased with cable and integral 220/240V power supply and comes with 10 ED diskettes.

We are exhibiting in:

LONDON:

18th January 1992
Royal Horticultural Hall
Greycoat Street,
Victoria

OTHER PRODUCTS:

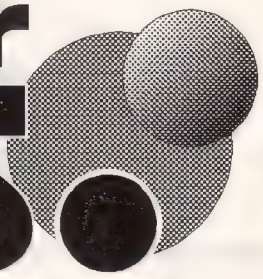
(more details about our products may be found in the QUANTA Magazine)

ITEM	PRICE	
QL CENTRONICS	£29.00	(£28.00 export)
QL 40M HARD DISK	£449.00	(£405.00 export)
* TRUMP CARD 768K	£100.00	(£89.00 export)
* TRUMP CARD 2	£150.00	(£130.00 export)
* EXPANDERAM 512K	£50.00	(£46.00 export)
5.25" DISK DRIVE	£95.00	(£88.00 export)
DISK ADAPTER	£15.00	(£15.00 export)

* reconditioned product

MIRACLE SYSTEMS Ltd, 25 Broughton Way, Osbaldwick, York, YO1 3BG, U.K. Tel: (0904) 423986

New Uses For ABACUS



PART ONE

Howard Clase believes that there is more to a spreadsheet than simply spreadsheeting.

Like many QL owners I suppose I routinely use *Quill* more than any other member of the Psion four, but I am always especially pleased when an opportunity to use *Abacus* comes along. As time goes on I find more and more uses for this very versatile program, although I don't go as far as the proverbial executives who regularly use their spreadsheets for word processing!

I have used Abacus for keeping the marks list for my classes, printing voting slips for a political meeting, printing planting instructions for bulbs, making up lab sheets, and, in conjunction with *Easel*, for calculating and plotting graphs of scientific functions. Yes, I know some of these sound suspiciously like word processing, but I can defend myself – in due course!

Several answers

To begin at the beginning, what exactly is a spreadsheet? There could be several answers to that question depending upon the user. To an accountant it is an electronic balance sheet – and this is where articles on Abacus in *Sinclair QL World* have generally stopped – but to a scientist it is a powerful calculation tool, to a teacher it is an electronic register and under special circumstances it can be better than Quill as a word processor. I should like to argue that at heart it is a programming language, a sophisticated editor and a memory monitor all rolled into one, and this is what makes it so useful in so many fields. Since most readers of QL World will be familiar with Basic, at least to some degree, let's compare the Abacus programming language with Basic, by means of a rudimentary spreadsheet in SuperBasic (see **listing one**).

If you type this in and run it you will find that it prints a 3 x 3 grid and does a couple of calculations in the manner of Abacus. The procedure 'Set_up' opens a set of small screen channels whose windows form the grid. The main program (lines 120 to 200) consists of a series of similar lines each containing two statements: the first defines a variable and the second displays its current value in an exclusive small window (or grid cell). You can edit the first statements to change the definitions, but any other changes destroy the structure of the program. Note that I have chosen variable names to correspond to the cell references, and the program executes the statements across the grid in the same order as you are reading this – from left to right across each row starting at the top and working down row by row. The definition may include the value of another variable but this obviously must have been defined earlier in the program – although, since RUN in SuperBasic does not CLEAR the variables, a couple of RUNs may, in fact, get everything right.

Abacus works in exactly the same way as my Basic program, but has many more sophisticated features. Since the program lines, variable names and cell positions are unalterably linked to each other, two of them are redundant: it is quite sufficient to use the cell position for all three, and that is what Abacus does. Executing Abacus (<F3>, <X>) is equivalent to RUN in SuperBasic, and it normally executes 'by rows', ie from left to right across each row from top to bottom of the grid (although there is the option to do it 'by columns' instead). Unlike Basic, however, there is no facility for jumping or looping; the structure is much more rigid and every cell is always evaluated – but there are ways around this; there is at least an IF function, but more of that later.

Listing one

```
100 nm$ =     "Abaemu_bas"

110 CLS: MODE 4: Set_up
120 a1 = 2:                 PRINT#1,a1
130 b1 = 3:                 PRINT#2,b1
140 c1 = 5:                 PRINT#3,c1
150 a2$="":                 PRINT#4,a2$
160 b2$="Sum = ":           PRINT#5,b2$
170 c2 = a1+b1+c1:          PRINT#6,c2
180 a3$="":                 PRINT#7,a3$
190 b3$="Product = ":       PRINT#8,b3$
200 c3 =a1*b1*c1:           PRINT#9,c3
210 STOP
220 REMark ^^^^^^^^^^^^^^^^^^^^^^^^^
230 DEFine PROCedure            Set_up
240 LOCal c,i,j,m,x,y: m=3
250 FOR j=1 TO m
260  FOR i=1 TO m
270   c= m*(j-1)+i: OPEN#c,con
280   x=128+64*(i-1): y=32+12*(j+1)
290   WINDOW#c,64,12,x,y
300   BORDER#c,1,4,0: CLS#c: INK#c,6
310 END FOR i: END FOR j: END DEFine
```

Emulator

The first problem with my little emulator is that in order to alter anything you have to use the QL's line editor. Note that you also have to be careful to add the $ sign to the variable name if you want it to contain text rather than numbers. With Abacus you use the arrow keys to move the cursor to the cell of interest and you are right in the editor; all you have to do is to type in or alter your definition. But you will still have to tell Abacus the 'type' of your entry if you want it to be text by typing in the double-quote character. A number of my correspondents have complained about this – saying that Abacus ought to be able to tell, but this is because they are not aware of the full potential of the program.

The problem is the availability of 'labels' for rows and columns, which enable a cell to be designated by a reference like 'July. Costs' as well as the grid reference eg C6.

If the program assumed that any word it did not recognise as a key-word must be text then you would not be able to use labels; it's the usual compromise. (That said, I personally, do not find labels particularly useful, and rarely use them myself; even if you do the program immediately converts them to the standard grid reference format anyway! The one use I have found for labels is for finding a piece of text in a large spreadsheet, eg a student's name in a register, if you press <F5> and then enter the text the cursor goes to the end of the first row containing the text.) Putting in the quotes is equivalent to having to put a $ at the end of the name of a string variable in SuperBasic. While you

should start with double quotes ("), Abacus will automatically supply the closing quotes. But if you are like me and often forget the initial quotes, all is not lost: you can put them in yourself – but now you must put both in, before and after (and you can use either single or double quotes as long as they match). You can convert a value or formula to 'text' too if you like by using the AMEND command <F3>, <A>; this is sometimes useful while building a spreadsheet, acting like a REM in Basic.

Double quote

If you do not start with the double quote Abacus assumes you are entering a numerical value or a formula. Typing a number assigns its value to that cell like a LET in Basic, while, if you want to put in a formula you have almost as large a range of functions available as in Basic. These can be combined into quite long expressions, the limit is about 160 characters. Just as in SuperBasic any formula typed in is checked for syntax before Abacus will accept it; the error message I get most often is 'undefined name reference', which in practice means that I have spelled something incorrectly!

If you are building up a large Abacus program it is a good idea to go into Design (<F3>, <D>) and press <A> to switch off the auto calculation. This stops it from running through the program every time you make an entry or alteration. If you need to test, then <F3>, <X> will do it for you when you need it.

To alter an existing definition then type <F3>, <A> this puts you directly into the Psion line editor which has more facilities than the one available when you are writing SuperBasic – <SHIFT+arrow> moves the cursor a word at a time, <SHIFT+CTRL+arrow> deletes whole words, and <CTRL + up/down-arrow> deletes from the cursor to the beginning/end of the line respectively. You will probably have become familiar with these extra facilities since they are also available in Quill.

Abacus acts like a software monitor in that it always displays the current 'value' of each cell in the grid, just like the monitors that machine code programmers use to examine the value of each byte of memory. In Basic terms there is an automatic PRINT statement in each cell. It is this current value as displayed on the screen that the program uses in its calculations when a cross reference is made, and not the formula, and it is the set of current values that make up the display that you will normally print out.

The equivalent of LISTing your Abacus program is definitely not as easy as with SuperBasic.

On screen the simplest way is to move the cursor through the grid cell by cell and see the contents of each cell printed at the bottom of the screen, but there is a problem, there is only room to display a single line (about 70 characters if you use a monitor, fewer in tv mode). So if you have entered a long formula you may not see it all. Fortunately, you can recover it all by using the AMEND command – <F3>, <A> –, which displays two whole lines of the contents of the current cell in the editing area of the screen. This may seem cumbersome, and you have to be in monitor mode to get all 160 characters, but it is the best available.

Getting a hard copy listing of your Abacus program is even more inconvenient. First, long formulae are not catered for; only one line is printed, truncated at 60 characters. (If anyone finds a way around this please let me know and I'll pass it on.) Second, the actual presentation of the formula is different from that which you have typed in or which appears on screen. This is a by-product of the relative addressing which is otherwise a strength of Abacus. If you refer to cell A1 while the cursor is on B3, and then COPY or ECHO the formula (<F3>, <C>/<E>) to, say, D4 you will find it now refers to C2 instead of A1. Abacus reads the reference as 'one column to the left and two rows up' rather than as an absolute reference to cell A1, and this is the way the reference is printed out – in this example as C[-1]R[-2]. This is because Abacus refers to a 'master' formula rather than a separate copy for each cell that uses the formula. If you select the formula option on printing then you will get a map of the grid with the location of each formula identified by a code number, eg F12, followed by a list of the formulae or the first 60 characters of them at any rate. (In the manual it says that the formulae come first, but my versions of Abacus all do it the other way round.)

Avoid clutter

Of course you can always write your programs with formulae shorter than 60 characters, but this may clutter up the screen with unnecessary intermediate values.

In the Abacus programs that follow I have printed the contents of a typical screen followed by a listing of the key formulae *as you would type them in*. Any errors should be attributed to my typing and not blamed on QL World's typesetters. All other cells contain either text or numerical values as seen on the screen.

But rather than continue in the abstract, let's look at a few actual examples of the sort of programs I am talking about. These are intended to introduce readers to some of the functions in Abacus that you may not meet in ordinary financial spreadsheet programming, and are examples rather than really useful programs in their own right.

Many numerical problems in science and maths can be reduced to feeding numbers into a standard formula – sometimes after a bit of algebraic massaging. Of course Abacus won't do the algebra for you, but it will do the calculation for you more reliably than your fingers on a calculator. Why use Abacus when you can write a SuperBasic program to do it? Because it is much simpler to set up Abacus to do this sort of thing since it is specifically designed to do calculations, and it is especially worthwhile if you want to do repeated calculations of the same type or to compare the results of several similar calculations. There is also the bonus of double-precision arithmetic; Abacus displays up to 14 digits against SuperBasic's 7. Don't be fooled though, the calculation is only as accurate as the measurements that are put in, and very few measurements are significant to 7 digits let alone 14!

Log rolling

There are far more mathematical functions available in Abacus than the average accountant ever finds use for, and even if they are fewer than those in SuperBasic they are carefully chosen so that the missing ones can be duplicated by combinations. For example, decadic logarithms are obtainable from the natural ones by LOG10(x) = LN (x)/LN(10). It is a pity that more attention wasn't given to consistency between programs – I always forget that the square root function is SQR(x) in Abacus and not SQRT(x) as in SuperBasic, you use CHR() and not CHR$() to convert Ascii codes to characters, and to concatenate strings in Abacus you must use + and not &. Also to give permission to overwrite a file when saving you must press <ENTER> and not <Y> as in Quill.

As a typical example I've been delving in my old school maths notes and found a formula for calculating the area of a triangle from the lengths of the three sides. As well as showing how to tackle this type of problem it illustrates the use of a couple of less familiar Abacus functions – 'ASKN' and 'IF'.

First look at the SuperBasic version of the program (**Listing two**). The three values are INPUT (150), and the value of the semiperimeter (s) calculated from them (160). This must be longer than any of the sides or they will not meet to form a triangle; line 170 tests this and only EXITs the loop if all is OK, otherwise there is an error message and a raspberry (180–190), and you go round again. (NB The value of the semiperimeter, s, is not printed on the screen.) Once you have got out of the loop the result is calculated and printed at line 210 (there is no need to assign the value to a variable before you print it).

Listing three shows an Abacus version of this program following the SuperBasic as closely as possible. The top of the listing shows the appearance of the screen after a typical run. The bottom half shows the contents of the cell as you would type

them in or as they appear in the command area when you move the cursor to the cell.

The function REPT (<character>,num) is like FILL$ in SuperBasic: it prints the character num times, in this example it is used to pretty up the screen. Since there is a blank character between each cell you must use width ( )+1 to get a continuous line without any gaps. The various messages are just straight text – it is useful to know that if your text is longer than the width of the cell it will run over into cells to the right as long as they are truly empty, so you don't have to break up the message cell by cell here.

Semiperimeter

ASKN is the Abacus equivalent to SuperBasic's INPUT, and must be used in association with execution of the program (<F3>, <X> – like RUN in SuperBasic). Once the three values are entered the value of the semiperimeter is calculated in C14, but because of the rigid structure of Abacus it has to be displayed whether you need to see it or not.

The next part of the program is different from the Basic one since there are no loops in Abacus. Fortunately there is an 'IF' statement available, albeit with a rather different syntax. Where in Basic we have:

IF <condition> THEN <action1> ELSE <action2>

in Abacus this becomes:

IF (<condition>, <value1>, <value2>)

that is, the cell contains value1 if the condition is true and value2 if it is false. In Abacus the result must always be a value (numeric or text), while in Basic a wider range of actions is possible.

The same inverted test is made in cell C15 as in line 170 of the SuperBasic version, but the result is that the text 'Error' appears if the sides do not meet and the empty string if they do. It has to be backwards since the Boolean function NOT is not available in Abacus. Fortunately AND and OR do work in the condition statement, despite not being mentioned in the manual.

The result of the test in C15 is used in cells B16 and C16. The explanation of the error is split between the two cells and only appears if there is an error. If there has been no error the final result appears in cell C17, otherwise Abacus will baulk at finding the square root of a negative number and give its own rather unhelpful error message.

A program like this could be set up for users uninitiated into the mysteries of Abacus. However, I find ASKN rather confusing in operation: the question appears at the bottom of the grid in the command line, and the value of the entry does not appear in the appropriate cell until all three have been entered, so you have no way of spotting a mistake until it is too late. For this reason I prefer the alternative version shown as 'Trgl_alt'. All the formulae are the same except that the ASKN functions in cells C10:C12 have been removed allowing direct entry of the lengths of the sides. You should also turn off the AUTO-CALCULATE function (<F3>, <D>, <A>, <ENTER>); this causes each value to appear as soon as it is entered; then when they are satisfactory press <F3>, <X> to execute the program. The values in this example have been chosen to illustrate the result of an error.

If you want to compare the results of several similar calculations side by side nothing could be easier, just COPY (<F3>, <C>, range) the working part (C10:C17 in our example) of the program to the adjacent column or columns. If the formulae contain absolute addresses you may have a bit of AMENDing to do – see **listing four.**

In the next article we'll look at some more examples.

Listing two

```
90 nm$ =          "Trgl_bas"

100 REMark  To calculate the area of a triangle
110 REMark from the lengths of the three sides.
120 REMark ^^^^^^ hjc 1991.06.25 ver 1 ^^^^^^^^
130 CLS
135 REPeat loop
140   PRINT 'Side a','Side b','Side c'
150   INPUT a,b,c
160   s=(a+b+c)/2
170   IF NOT(s<a OR s<b OR s<c): EXIT loop
180   PRINT 'Error - sides do not meet'
190   BEEP 2500,25
200 END REPeat loop
210 PRINT \\'Area ='!SQRT(s*(s-a)*(s-b)*(s-c))
```

Listing three

```
          <<<<  Trgl_aba  >>>>

   |A|        B        |        C
  1|  ** The area of a TRIANGLE  **
  2|  from the lengths of the sides.
  3|
  4|  ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  5|  >>> Press <F3>,<X>, and enter the
  6|  lengths of the sides as requested
  7|  at the bottom left of the screen.
  8|  ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  9|
 10|  side a                          41
 11|  side b                          40
 12|  side c                           9
 13|
 14|  semiperimeter, s                45
 15|
 16|  ================================
 17|  Area =                         180

              Key Formulae

B4: rept("^",width()+1) Echoed to C4,B8,C8
C10: askn(">>>>> SIDE a ")
C11: askn(">>>>> SIDE b ")
C12: askn(">>>>> SIDE c ")
C14: sum(C10:C12)/2
C15: if(C14<C10 or C14<C11 or C14<C12, "Err
     or","")
B16: if(C15="Error","* Sides do not f",rept
     ("=",width()+1))
C16: if(C15="Error","orm a triangle. *",rep
     t("=",width()+1))
C17: sqr(C14*(C14-C10)*(C14-C11)*(C14-C12))

All other cells contain text as seen above.
```

Listing four

```
          <<<<  Trgl_alt  >>>>

   |A|        B        |        C        |D
  1|  ** The area of a TRIANGLE  **
  2|  from the lengths of the sides.
  3|
  4|  ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  5|  <<<    Move  the  cursor  to  the
  6|  appropriate  cell (C10 - C12) and
  7|  enter the length of the side. <<<
  8|  ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  9|
 10|  side a                          19 <<<
 11|  side b                           8 <<<
 12|  side c                           7 <<<
 13|
 14|  semiperimeter, s                17
 15|                  Error
 16|  * Sides do not form a triangle. *
 17|  Area =           ##ARG
```

SINCLAIR QL WORLD

THE NEW USERGUIDE

SECTION ELEVEN

KEYWORD INDEX

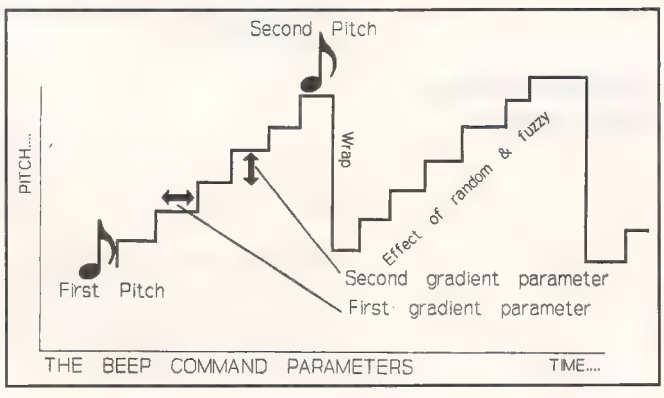
This month in the Keyword Index, Mike Lloyd moves from BEEP to CLOSE in SuperBasic, skirting Super Toolkit 2 and Turbo Toolkit on the way.

BEEP (duration, pitch1, pitch2, grad1, grad2, wrap, fuzzy, random)

	SOUND COMMAND
duration (0 to 32767)	the length of the sound. 1 = 72 microseconds
pitch1 (0 to 255)	the first note pitch
pitch2 (0 to 255)	the alternate note pitch
grad1 (-32768 to 32767)	time interval between gradient steps
grad2 (-8 to 7)	pitch difference between steps
wrap (0 to 15)	number of gradient repetitions
fuzzy (0 - 15)	distorts pure tone
random (0 - 15)	random element affecting all parameters

The BEEP command can cause more hair-loss than almost anything else associated with the QL. Bearing in mind that the QL was supposed to be a business machine, its sound capabilities are surprisingly oriented towards arcade games. It is impossible to play a simple tune without a great deal of effort because the pitch values are not related to musical tones and semitones. However, if the sound of a rampaging, homicidal alien is required, the QL can quickly come up with suitable suggestions.

Without any parameters, the BEEP command performs the valuable function of stopping any noise currently in progress. The minimum parameters acceptable are a duration and a pitch. A one second note (of indeterminate pitch) is produced by the command BEEP 720, 8. Note that a permanent note is obtained by giving a duration of 0, which also has the unfortunate effect of changing the pitch of the note. If a second pitch is given you must also give the speed with which the QL will 'bounce' between the two notes and the pitch difference between each tone in the gradient. Wrap, fuzzy and random are further, optional parameters which distort the sound in ways best listened to rather than described.



BEEPING ( )

SOUND FUNCTION

(No parameters – brackets are optional)

The BEEPING function returns True if the QL's sound chip is working and False if it is not. You may not assign these values to a variable (BEEPING will return a 0 if you try) so BEEPING is restricted to IF and SELECT statements. Purists will want to put brackets at the end of the function, but those less fussy can get by without them.

BGET #chan \offset, byte1, byte2 . . .

[SUPER TOOLKIT 2]

LOW LEVEL CHANNEL ACCESS

#chan	a valid channel number, usually a file
offset	an optional value denoting where in the file to begin reading data expressed as a number of bytes from the beginning of the file.
byteX	a value between 0 and 255.

BGET (B stands for Byte) reads bytes from a channel normally associated with a file. BGET assumes that the file is simply one long string of bytes. The following command will read the Ascii values stored between character positions 20 and 22:

```
BGET #3 \20, A, B, C
```

Whether the results are of value depends upon the structure of the file being read.

BIN$ (decval, bits)

[SUPER TOOLKIT 2]

BASE CONVERSION FUNCTION

decval	an integer value
bits	the number of binary digits to produce

BIN$ takes a decimal value and converts it into a string of 1s and 0s which represent its binary value. To see directly the result of a binary AND, use the following commands:

```
PRINT BIN$ (137, 8)
PRINT BINS$ (89, 8)
PRINT BIN$ (137 && 89, 8)
```

BIN (string$)

[SUPER TOOLKIT 2]

BASE CONVERSION FUNCTION

STRING$	any string of characters, but normally made up of 1s and 0s.

BIN converts any string of characters into a decimal number by assuming that characters with even Ascii values represent 0 and those with odd Ascii values represent 1. Conveniently, the Ascii values of 0 and 1 are even and odd respectively.

BLOCK #chan, width, height, xpos, ypos, ink

PIXEL-BASED GRAPHICS COMMAND

#chan	A screen channel
width	The width of the block
height	The height of the block
xpos	The lateral co-ordinate of the block's top left corner
ypos	The vertical co-ordinate of the block's top left corner
ink	The colour of the block (0 - 255)

BLOCK is unique in that it is the only graphics command to make use of the pixel co-ordinate system starting at the top left corner of each window. Because it does not need to translate between arbitrary floating point graphics co-ordinates and pixel positions it is significantly faster than any other graphics command. Use it to draw vertical and horizontal lines in preference to the conventional LINE command. If OVER is set to -1 a BLOCK of a suitable colour covering the whole window will recolour the window very much faster than RECOL can manage, but with less control over the colours achieved. Unlike conventional graphics commands, BLOCK causes errors if you place any part of the block outside the window.

BORDER #chan, width, paper

GRAPHICS COMMAND

#chan	(optional) A screen channel
width	The width of the border in pixels
paper	(optional) The colour of the border (0 - 255)

Every window can have a border around it, reducing the room available to the active screen. The border's width remains constant whether a screen is displayed in high or low resolution. If no 'paper' value is given the border is transparent, therefore a multi-coloured border can be obtained by specifying the innermost colour first and ending with a wide, transparent border to protect the colours, as in:

```
BORDER 8, 5
BORDER 6, 3
BORDER 4, 1
BORDER 8
```

BPUT #chan, byte, byte, . . .

[SUPER TOOLKIT 2]

LOW LEVEL CHANNEL ACCESS

#chan	a valid channel number, usually a file or the printer
offset	an optional value denoting where in the file to begin reading data expressed as a number of bytes from the beginning of the file.
byte	a value between 0 and 255.

Information is passed to and from QL devices, such as the screen, printer and files, one byte at a time. When you think that you are printing the word HELLO on your screen the computer is actually transmitting 72, 69, 76, 76 and 79 to the channel associated with the default window. These figures are the Ascii equivalent of the characters HELLO. The command BPUT 72, 69, 76, 76, 79 is the low-level equivalent of typing PRINT "HELLO". The command is of more value when storing values in a file or selecting printer settings. Setting a five-character-wide left margin on an Epson printer can be achieved by either of the following commands:

```
PRINT #5, CHR$(27); "l"; CHR$(5): REM 28 significant keypresses
BPUT #5, 27, 108, 5: REM 15 significant keypresses
```

CALL address, data1, data2. . . data 13

MACHINE CODE PROGRAM COMMAND

address	A valid, even-numbered memory location.
dataX	(optional) Four-byte integer values.

CALL is used to initiate a machine-code program previously loaded into a reserved part of the QL's memory with a RESPR command and an LBYTES command. The data parameters are loaded into the cpu addresses D1 to D7 and A0 to A5 prior to the program beginning.

CDEC$ (value, width, decplaces)

[SUPER TOOLKIT 2]

DECIMAL NUMBER FORMATTING COMMAND

value	An integer value
width	The total number of character positions to return
decplaces	The number of decimal places to display

CDEC$ is an extremely useful way of circumventing the QL's habit of placing numbers with relatively few significant digits into exponential format. As a bonus it even provides fixed length strings and separates thousands with commas for attractive, justified columns of figures. Although CDEC$ output can include decimal places, only the integer part of the input value is recognised. This means that you must think of £34.25 as 3,425 pennies. The results can be observed with this snippet:

```
100 REPeat loop
110 pennies = RND (999) * RND (999)
110 PRINT "£"; CDEC$ (pennies, 8, 2)
120 END REPeat loop
```

CHANNEL _ ID (#chan)

[TURBO TOOLKIT]

CHANNEL FUNCTION

chan	An open SuperBasic channel number

SuperBasic and Qdos do not agree over what a particular channel is called. This is so that the QL can support multi-tasking: one program might have Channel #3 linked to a printer at exactly the same time that another program has Channel #3 linked to a screen window. Qdos must be able to service all channel requests and so gives each channel a unique number normally invisible to programmers. *Turbo Toolkit* provides this simple function to return the Qdos channel number for any given SuperBasic channel.

CHARGE (task name)

[TURBO TOOLKIT]

COMPILER DIRECTIVE

filename	(optional) A valid taskname

CHARGE launches a Digital Precision compiler (which one depends upon your default setup). If a taskname is included it is used as the name of the task being compiled.

CHAR _ INC #chan, X_ inc, Y_ inc

[SUPER TOOLKIT 2]

CHARACTER COMMAND

#chan	(optional) A screen channel
X_inc	lateral spacing of characters in pixels
Y_inc	vertical spacing of characters in pixels

SuperBasic offers very limited options for sizing and spacing characters: many of these restrictions are removed by *Super Toolkit 2*. Should you want characters printed at 14 pixel intervals on lines 15 pixels apart, issue the command CHAR_INC 14, 15.

CHAR_USE #chan, font1, font2

[SUPER TOOLKIT 2]

CHARACTER COMMAND

#chan	(optional) A screen channel
font1	the start address of the first font
font2	(optional) the start address of the second font

Qdos divides the full character set into two fonts. Super Toolkit 2 allows you to specify different fonts in each of the screen windows should you wish. Each font is loaded into ram at reserved addresses and the CHAR_USE command is issued to make Qdos aware that the new font is to be used by a specified window. Should you accidentally provide an inaccurate address, all characters will be replaced by garbage. To reset the fonts to their default designs held in the QL's rom, simply issue the command CHAR_USE 0, 0.

CHR$ (ASCII_code)

CHARACTER COMMAND

ASCII_code	An integer between 0 and 255

The Ascii code assigns specific characters to the values 0 to 127, leaving the values 128 to 255 for individual system designers to assign. Thus, every component of a computer's character set can be represented by single byte. The first 32 Ascii codes are non-printable codes representing such functions as backspace, newline and tab. CHR$ is a function which takes a one-byte value (ie an integer between 0 and 255) and produces the character which that number represents. Be careful with the values 0 to 31 because they may produce unexpected results. Minerva owners, however, can obtain printable characters from these values. Super Toolkit 2 owners will prefer to use BPUT to CHR$ wherever possible.

CIRCLE #chan, xpos, ypos, radius, distort, angl, [xpos, ypos, etc]

CIRCLE #chan, xpos, ypos, radius, distort, angl, [xpos, ypos, etc]

GRAPHICS COMMAND

#chan	(optional) channel number
xpos, ypos	co-ordinates of circle centre
radius	radius of circle in graphics units
distort	(optional) ratio between major and minor axes of an ellipse (0 to 1)
angle	(optional) orientation of major axis in radians (0 = vertical)

CIRCLE uses absolute co-ordinates; CIRCLE–R uses relative co-ordinates. CIRCLE draws circles and ellipses on the screen in the current INK colour using the graphics co-ordinates system. IF FILL 1 is in effect the circles will be solid. A bug causes 'colour leaks' if FILL 1 is not issued between successive CIRCLE commands. By separating sets of parameters by semi-colons a single CIRCLE command can draw many circles. The CIRCLE and ELLIPSE commands are absolutely identical. A simple but attractive example of circles is:

```
100 FOR x = 1 TO 100: CIRCLE 50, 50, x*2, 0.3, x/20*PI
```

CL CHIP

MEMORY MANAGEMENT

The QL's memory manager can allocate memory from the 'common heap' for use as reserved memory areas. Qdos also grabs some of this space to record microdrive and disk information (which is why the QL takes seconds to examine a microdrive on the first access and subsequently is content on subsequent accesses to affirm that the drive contains the same cartridge). CLCHP is short for CLear Common HeaP, and it does just that. Use this command with care.

CLEAR

MEMORY MANAGEMENT

When a SuperBasic program runs on the QL it needs memory space to record variable values, etc. CLEAR removes all trace of variables from the QL's memory. Every time a procedure is called, the QL memorises the current state of variables which might have local assignments within the procedure. If a program is halted in the middle of a user-defined procedure or function call the computer still believes it is in the user definition. CLEAR clears this misunderstanding.

SOFTWARE FILE

SQUIDGY ROUND THE WORLD

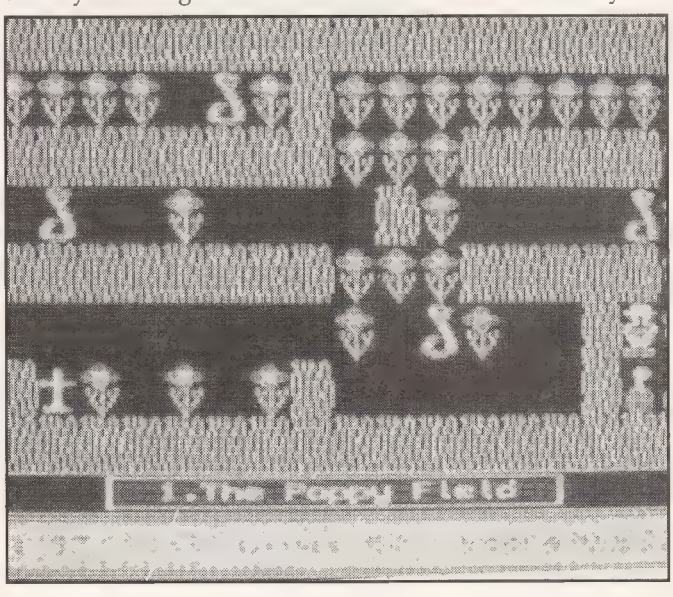
INFORMATION
Program: *Squidgy round the World*
Supplier: CGH Services, Cwm Gwen Hall, Pencader, Dyfed, Wales SA39 9HA.
Price: £10.00 (plus £1.00 p&p) on 3.5 or 5.25 disk. £12.00 (plus £1.20 p&p) on two mdvs. (Both must be supplied, formatted to 220 sectors.)

John Shaw finds this 50-screen arcade game anything but soft and damp, with excellent graphics.

Bored of *BJ*? Tired of *Tetris*? Beaten by *Brainsmasher*? Then here we are, a new arcade game from the stable of GCH Services and it's all about a friendly chap called Squidgy. The story goes thus:

'*Squidgy* came from a gravel pit in the Cotswold Lakes. He was given that name on account of his strange skin. Various interested parties around the world want to perform experiments on him thinking he is a relative of the Loch Ness Monster or the Yeti.'

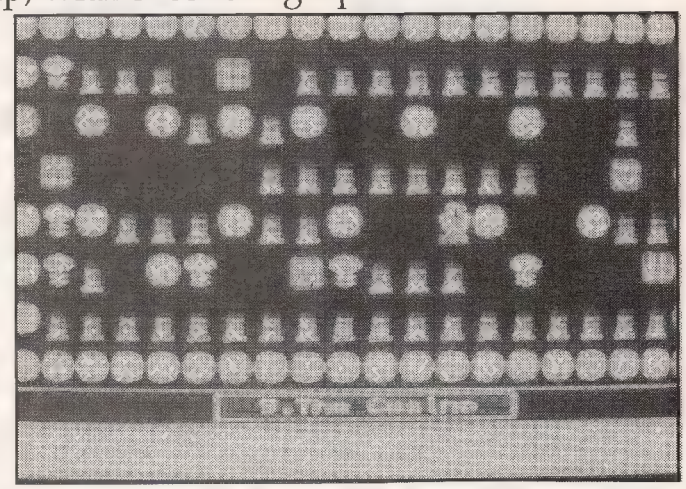
After booting up, the first scenario will show a poppy field. Avoid the snakes and pick the poppies. Once you have picked all of them you go to the exit. At most places you exit by walking off of the playing area.



Bombs! Avoid these at all cost as they are very temperamental; the smallest nudge could result in a nasty headache and they feature in every screen.

Some locations look impossible to complete but there is a little cross hidden somewhere which gives you limited immortality. You can walk through three to five of your enemies before dying. Bombs always kill you. A little tip: you can also move diagonally – it's the only way through some screens!

The Jumping Beans at some places are extra lives - very useful! Clocks increase your bonus time so you are more likely to get a bonus when you complete the place. If you forget what you are supposed to be collecting then press the backslash (\) key which doubles as a pause key. If the various bleeps and buzzes annoy you then you can press the pause button and TAB which will enable/disable sound.



If you are very good and you visit all of the places (very unlikely unless you have the reflexes of a panther) then you will restart at level one but with increased speed.

When you have lost all of your lives the game will finish and if your score was high enough you will be able to enter your name on the high-score table.

There is a cautionary note: avoid running Squidgy on microdrives if you use a Minerva rom that changes the address of system variables. It will run perfectly under Minerva from any other device.

The playing format is a well tried one but it loses nothing for all that. The author has put a lot of work into the graphics and the whole thing runs smoothly and well.

More difficult

I found it a very enjoyable game which I think is particularly suited to the younger element of the family. Don't be fooled by the ease of the initial few screens. There are 50 of them in all and they get more difficult as you progress.

Good value for money and an excellent birthday present for your children.

DIY TOOLKIT

Simon Goodwin extends SuperBasic with menu-driven editing functions and improved Name Table access.

Among other treats, this DIY Toolkit instalment provides a 'menu-driven front-end' for SuperBasic program editing. It works rather like the procedure editor in Psion's *Archive,* and was suggested by keen *QL World* reader Anders Hartzellius of Sweden, who sent a prototype of the code. Anders wins the pick of DIY Toolkit disk volumes.

All the names

The functions _DEF% and _DEF$ display a menu of all the Basic procedure and function names defined in your program. You can move a highlight through the list and select any definition by pressing Space or Enter. The list scrolls if there are more names than will fit in the chosen window.

As usual for DIY Toolkit, the functions are fast and reliable on all QL rom versions. They adjust automatically to suit all Modes, window positions and character sizes.

_DEF% works best with the ED screen editor command from Sinclair's *QL Toolkit* or Care's *SuperToolkit 2,* or the equivalent in ABC Elektronic's extravagantly named *Gigatoolkit.* You can also use it with the standard EDIT command, but if so you may like to change the default channel from #2 to #0.

Underscore

Anders' choice of function names uses a leading underscore to prevent clashes with existing SuperBasic names. You can easily change them with DIY Toolkit's ALIAS command if you don't like typing the underscores. The difference between _DEF% and _DEF$ is that _DEF% returns the line number of the chosen routine, while _DEF$ returns the name, as a string. Both functions accept one optional parameter, which should be the number of a SuperBasic Con channel. Over 0 is assumed, but this is no limitation as ED and EDIT have the same requirement.

The menu produced by _DEF saves you scrolling through the program looking for the required definition. Procedures and functions make programs easier to read, not least because they allow you to refer to routines by a meaningful name of your choice. Used on their own, ED and EDIT are a retrograde step as they require you to remember the line numbers as well as the name. Numbers are often confused and easily mistyped.

You could use Toolkit commands like QFIND, LDEF or LOOKUP% to find a line from the name of a definition, but the menu approach is often preferable. It displays all the possibilities together. There is little risk of typing mistakes or mistakenly requesting the first thing to come into your head.

Altkey

ED _DEF% is not too arduous a command, but since you will use it every time you want to edit a SuperBasic procedure or function it makes sense to define it as an ALTKEY that can be called up with a single stab at the keyboard. This Toolkit 2 command calls up the menu, and the Toolkit editor, when you press ALT and the letter 'e' at the same time:

```
ALTKEY 'e', 'ED _DEF%' & CHR$ (10)
```

If you find ALT 'e' hard to find or remember, replace the letter 'e' with some other character of your choice. The ALT key is in the right corner, near Enter.

By default the menu of names appears in the listing window #2, but you can direct it elsewhere by adding the channel number in brackets after the function:

```
EDIT _DEF% (#0)
```

If the window is very narrow or the names are very long they are trimmed to fit. The standard windows in F2 (TV) mode show the first 36 characters of long names. _DEF$ always returns the full name chosen.

One direct

The functions report 'not found' if there are no SuperBasic procedure or function definitions. If there is only one candidate name the functions select it automatically, as there is little point in presenting a menu with only one option. You can break out of the menu by pressing Ctrl Space or Esc. The same keys halt ED, but you need Minerva to interrupt EDIT with Esc.

The routines behave predictably if there are duplicate definitions, or you change a name by editing the program. Imagine a procedure called CAT, renamed CATALOGUE later in the editing session. Both names appear in the _DEF menu, and you end up at the same line whichever one you pick.

One definition

SuperBasic's Name Table only keeps details of one definition for each name. If you use the same name twice the menu option selects the most recent definition loaded, entered or merged.

If a definition is deleted, or over written by MERGE so that new lines replace it, the name still appears in the menu; if selected, _DEF% returns the line where the DEF used to be. Such antique names are cleared out if you SAVE and re-LOAD the program. Alternatively you can cancel them with DIY Toolkit commands.

If NP is the index number of the deleted name, found with LOOKUP% or BASIC _INDEX%, you can remove it from subsequent menus like this:

```
BPOKE_W BPEEK_L (24) + NP * 8, 2
```

This changes the type in the Name Table to 'unset number'. If you accidentally cancel a name that still exists in your program, you can restore it to the menu by editing and re-entering the DEFine statement.

Secret vector

If your program has short lines and many procedures and functions you may wish to expand the Buffer to make room for more lines. _DEF% and _DEF$ will automatically use the extra space, like MORE, INPUT and COPY. You can expand Buffer with an undocumented Qdos vector which works on all roms and emulators:

```
CALL PEEK_W (282) +28, 1000
```

I shall explain this, and other secret vectors, in my next DIY Toolkit column. For now, all you need to know is that the second parameter is the number of extra bytes required in the Buffer. For all but the most enormous programs, 1000 should be enough as it makes room for at least 500 name indices.

You may not find the menu useful when you want to edit the main body of your program – the few lines that call the procedures and functions that do the real work. These should come at the start, so you can get straight to them with the command ED, without parameters.

At first sight _NAME$ is just a concise synonym for *Turbo Toolkit's* BASIC_NAME$, but it incorporates differ-

```
* QL World DIY TOOLKIT _DEF%, _DEF$, _NAME$ functions, version 1.5
* (c) Anders Hartzelius 1991. DIY changes (C) 1991 Simon N Goodwin
*
sd_extop        equ     $09             Extended operation
sd_nl           equ     $12             New line trap key
sd_clear        equ     $20             CLS trap key
err_no          equ     -6              NOT OPEN error code
err_nf          equ     -7              NOT FOUND
err_bp          equ     -15             BAD PARAMETER
bv_tkbas        equ     $08             Offset to beyond Buffer
bv_ntbas        equ     $18             Offset to Name Table
bv_ntp          equ     $1c             Limit of Name Table
bv_nlbas        equ     $20             Offset to Name List
bv_chbas        equ     $30             Offset to Channel Table
bv_chp          equ     $34             Limit of Channel Table
bv_rip          equ     $58             Limit of RI Stack
bp_init         equ     $110            Basic extension vector
ca_gtint        equ     $112            Vector to get integers
*
start           lea.l   define,a1       Load address of definitions
                movea.w bp_init\w,a2    Fetch the ROM vector word
                jmp     (a2)            Initiate the new functions
*
name            moveq   #0,d5           Clear high bytes
                bsr.s   get_an_int      Pick up the parameter
                lsl.w   #3,d5           Multiply 0-8191 by 8 fast
                movea.l 32(a6),a4       Find the Name List offset
                movea.l 24(a6),a5       Find the Name Table offset
                adda.l  d5,a5
                cmpa.l  28(a6),a5       Check against end of Table
                bmi.s   check_list
bad_index       moveq   #err_bp,d0      Bad parameter, no such name
                rts
check_list      move.w  2(a6,a5.l),d1   Pick up Name List vector
                addq.w  #1,d1           Check it records a name
                beq.s   bad_index
                moveq   #1,d7           Indicate the result type
                bra     stacker
*
get_an_int      movea.w ca_gtint\w,a2   Yes: fetch channel number
                jsr     (a2)            Get integer
                bne.s   deep_err            Error ?
                subq.l  #1,d3           One parameter ?
                beq.s   par_ok
bad_param       moveq   #err_bp,d0      Bad parameter
deep_err        bra     quit_out        Report to basic
par_ok          move.w  0(a1,a6.l),d5   D5:=parameter
                bmi.s   bad_param
                addq.l  #2,bv_rip(a6)   Tidy RI Maths Stack
err             rts
*
defn            moveq   #3,d7           Flag type for  DEF%
                bra.s   def
defs            moveq   #1,d7           Flag type for _DEF$
def             swap    d7              D7 is used for two purposes
                moveq   #2,d5           Default channel
                cmpa.l  a3,a5           Any parameters ?
                beq.s   default         No: use default channel
                bsr.s   get_an_int      Check for one integer param
default         moveq   #err_no,d0      Posit Channel not open
                mulu    #40,d5          D5:=offset in channel table
                add.l   bv_chbas(a6),d5 Add base of channel table
                cmp.l   bv_chp(a6),d5   D5 -> past end ?
                bge.s   err             Yes: report error
                move.l  0(a6,d5.l),d5   D5:= channel id
                bmi.s   err             Channel closed ?
*
* Prepare the menu window
*
                move.l  d5,a0           A0:= channel id
                moveq   #-1,d3          Time out
                moveq   #sd_clear,d0
                bsr     trap3           CLS
                lea.l   get_csizes,a2   Load address of code
                moveq   #sd_extop,d0    Call extended operation
                bsr     trap3
                move.w  d1,d7           D7.W := width in chars
                move.l  d1,d4           D4.HW := Char. height
                lea.l   get_colours,a2  Load address of code
                moveq   #sd_extop,d0    Call extended operation
                bsr     trap3
                move.l  d1,d5           D5:= ink & strip
*
* Point at SuperBASIC tables and work out the buffer size
*
                movea.l bv_ntbas(a6),a5 Base of name table
                movea.l bv_ntp(a6),a3   End of name table
                movea.l bv_nlbas(a6),a4 Base of name list
                movea.l (a6),a2         Base of buffer
                move.l  bv_tkbas(a6),d6 D6 -> Limit of buffer
                sub.l   a2,d6           D6 is buffer size (bytes)
                lsr.l   #1,d6           D6 is buffer size in words
*
* Scan the Name Table and display SuperBASIC PROC and FN names
*
                clr.w   d4              Reset Window full flag
nt_loop         addq.l  #8,a5           Next entry in name table
                cmpa.l  a5,a3           End of name table ?
                beq.s   end_nt
                move.b  0(a5,a6.l),d1   D1:= name type
                subq.b  #4,d1           SuperBASIC Procedure ?
                beq.s   proc_fn
                subq.b  #1,d1           SuperBASIC Function ?
                bne.s   nt_loop
proc_fn         subq.l  #1,d6           Room for one proc / fn less
                beq.s   end_nt          Don't overflow the buffer
                move.l  a5,d1
                sub.l   bv_ntbas(a6),d1 Relative to start of Table
                move.w  d1,0(a2,a6.l)   Buffer offset in name table
                addq.l  #2,a2
                tst.w   d4              Test window full flag
                bne.s   nt_loop         No printout if set
                bsr     print_name
                moveq   #sd_nl,d0       New line
                bsr     trap3
                beq.s   nt_loop         Jump if not ERR_OR
                addq.w  #1,d4           Window full
                bra.s   nt_loop         Next entry in NT
*
end_nt          movea.l (a6),a3         Anything in the buffer ?
                cmpa.l  a3,a2
                bne.s   ok
                moveq   #err_nf,d0      Not found
                rts                     Quit if no proc/fn
ok              lea.l   2(a3),a1        Perhaps there's only 1 ?
                cmpa.l  a1,a2
                bne.s   choose          If more, use the menu
                moveq   #0,d1           Prepare for unsigned maths
                move.w  0(a3,a6.l),d1   D1:= index in name table
                add.l   bv_ntbas(a6),d1
                movea.l d1,a5           A5 is offset inside BASIC
                bra     done            Return the only name known
```

QL World DIY TOOLKIT. January 1992. Listing 1. page 2 of 2

ences and improvements which make it suitable in different circumstances.

BASIC_NAME$ was designed to read the Name List of SuperBasic Task 0, 0 from any task. _NAME$ differs by looking in the local list, suiting Minerva's extra interpreters and *QLiberator* (unless you use the NONAMES option to save memory). *Turbo* and *Supercharge* tasks do not contain a Name List, so they invariably report 'bad parameter' if used to call _NAME$.

Obscure bug

There is a very obscure bug in BASIC_NAME$, which does not affect compiled tasks or the Runtime Toolkit, but could cause problems if a task stops or starts at the wrong moment while BASIC_NAME$ is being interpreted. The _NAME$ function has no such bug; it also rejects parameters that point at the Name Table but do not correspond to entries in the Name List.

BASIC_NAME$ gives a string of gibberish, typically mixing names like PRINT and INPUT with control codes, if you inadvertently pass it the index number of an 'expression' entry at the end of the Name Table. _NAME$ spots that there is no corresponding name, and reports 'bad parameter' instead.

Number of names

_NAME$ expects a single integer parameter with a value between 0 and 8191. In practice the highest valid parameter depends on the number of names in the current program and loaded as extensions. Every QL rom includes at least 113 resident names, ranging from PRINT to VER$.

The SuperBasic Name Table is limited to 8192 entries, as each consists of eight bytes and the Name List grows to match. Entries in the List are found via word offsets in the Name Table which can address up to 64K of data, and 64K/8 = 8192.

The code in **Listing one** is derived from Anders Hartzellius's original, updated by Simon N Goodwin. The version Anders sent worked well enough to demonstrate the utility of _DEF%, but it had characteristic bugs that often afflict QL programs. I hope I have not overcomplicated the original design in the process of fixing it.

Complete source, assembled code and documentation for these functions has been added to DIY Toolkit Volume A, where it joins ALIAS, CODEVEC and INVERSE. The single volume costs £7. 18 volumes are available from **DIY Toolkit, Cwm Gwen Hall, Pencader, Dyfed Cymru SA39 9HA, tel: 0559 384574.** Each volume costs £3, plus £4 per order to cover disks, post and processing. Please enclose one cartridge per volume if ordering microdrive copies. Bargain Bundles are now available, costing £20 for six volumes on disk, with *SuperBasic, Devices* and *Utility* themes. Anders is working on his own update to_DEF$ and _DEF%, which will appear in the Swedish QL Users' Group magazine.

Quick entry

Listing two is a quick way to enter the code without using an assembler. It loads

the equivalent machine code from DATA statements, and saves the code in a file. Once you've loaded that file, as follows, you can use _DEF%, _DEF$ and –NAME$ in your own programs.

base=RESPR (626)
LBYTES "filename", base
CALL base

The first part of Listing two is the standard loader used in every month's DIY Toolkit project. Only the DATA, from line 590 onwards, changes for each set of extensions.

Customise

Listing one was written and assembled using HiSoft's *Devpac* and the Computer One assembler. Type this text into your own assembler if you want to customise the code. I have added 'A' to some instructions to placate pedantic assemblers like Metacomco's ASM, but you may need to add similar tweaks if your program does not recognise 'generic' Motorola opcodes.

The listing starts with a set of 'equates' or symbolic names for constants used later. Anders supplied a full set defined using hexadecimal values – hence the $ prefixes. More equates will be published next issue with the remainder of the listing.

Quirk

The START routine includes a Devpac quirk, the '\W' after word vector values. Most assemblers take that for granted, so it can be missed out, but Hisoft need it to generate concise code – otherwise Devpac uses a long address, wasting two bytes.

Next comes the code for _NAME$. The subroutine GET_AN_INT reads the parameter word, which is multiplied by 8 and used to find a Name Table entry. If the second word is -1 there is no name text; otherwise the routine returns the name as a string, using STACKER, part of the _DEF$ code.

Difference

The start of DEF$ and DEF% differ only in the 'result type' code loaded into register D7 and DEFN and DEFS. The program swaps this into the top word of the register, so D7.W can be used later to record the window width.

If A3 and A5 match, the default channel #2 is used, otherwise GET_AN_INT is called to fetch and check the parameter, before looking it up in SuperBasic's Channel Table. A0 is the identifier of a Con channel, which is blanked and then interrogated with two EXTOP calls. The high word of D4 becomes the character height in pixels, and the halves of D5 hold the INK and STRIP colours. Anders likes D5.

```
100 REMark Sinclair QL World HEX LOADER v 3
110 REMark by Marcus Jeffery & Simon N Goodwin
120 :
150 CLS: RESTORE : READ space: start=RESPR(space)
160 PRINT "Loading Hex..." : HEX_LOAD start
170 INPUT "Save to file...";f$
180 SBYTES f$,start,byte : STOP
190 :
200 DEFine FuNction DECIMAL(x)
210 RETurn CODE(h$(x))-48-7*(h$(x)>"9")
220 END DEFine DECIMAL
230 :
240 DEFine PROCedure HEX_LOAD(start)
290 byte = 0 : checksum = 0
300 REPeat load_hex_digits
310    READ h$
320    IF h$="*" : EXIT load_hex_digits
330    IF LEN(h$) MOD 2
340       PRINT"Odd number of hex digits in: ";h$
350       STOP
360    END IF
370    FOR b = 1 TO LEN(h$) STEP 2
380       hb = DECIMAL(b) : lb = DECIMAL(b+1)
390       IF hb<0 OR hb>15 OR lb<0 OR lb>15
400          PRINT"Illegal hex digit in: ";h$ : STOP
420       END IF
430       POKE start+byte,16*hb+lb
440       checksum = checksum + 16*hb + lb
450       byte = byte + 1
460    END FOR b
470 END REPeat load_hex_digits
480 READ check
490 IF check <> checksum
500    PRINT"Checksum incorrect. Recheck data.":STOP
520 END IF
530 PRINT"Checksum correct, data entered at: ';start
560 END DEFine HEX_LOAD
570 :
580 REMark Space requirements for the machine code
590 DATA 626
600 :
610 REMark Machine code data
620 DATA "43FA02503478011","4ED27A006124E74D"
630 DATA "286E00202A6E0018","DBC5BBEE001C6B04"
640 DATA "70F14E753236D802","524167F47E016000"
650 DATA "01A43478011124E92","66065383670670F1"
660 DATA "6000017A3A31E800","6BF454AE00584E75"
670 DATA "7E03600027E014847","7A02BBCB670261D2"
680 DATA "70FACAFC0028DAAE","0030BAAE00346CDE"
690 DATA "2A3658006BD82045","76FF7020610000132"
700 DATA "45FA01B8700961000","01283E01280145FA"
```

Identity

The next step is to identify all the SuperBasic procedure and function names. This involves loading address registers with internal table offsets. A5 and A3 point to the start and limit of the Name Table inside Basic. A4 points at the Name List, and A2 points at the start of the Buffer.

Buffer sizes vary from 128 bytes to 32K or more. Each name on the menu needs a word, so I calculate the number of words available and store it in D6. The low word of D4 is used as a flag, initially zero while the menu is drawn.

The program prints names of type 4 (DEF PROC) and type 5 (DEF FN) to the screen as they are found, until an 'ERR_OR' out of range error occurs when SD_NL is called to move to a new line. The error is trapped and D4 is set to one, suppressing further output, but Name Table offsets continue to be stored in the Buffer until the end of the table is reached or D6 counts down to zero, indicating that the buffer is full.

```
710 DATA "019C70096100011A","2A012A6E00018266E"
720 DATA "001C286E00202456","2C2E00089C8AE28E"
730 DATA "4244508DB7CD6730","1235E8005901670 4"
740 DATA "530166EE53866720","220D92AE00183581"
750 DATA "E8005 48A4A4466DA","610000BA7012610 0"
760 DATA "00D067CE52446 0CA","2656B5CB66047 0F9"
770 DATA "4E7543EB0002B5C9","6610720033233E800"
780 DATA "D2AE00182A416000","00C8720074007010"
790 DATA "6100009E48447200","3233E800D2AE0018"
800 DATA "2A4161000 09E616C","6100009870016100"
810 DATA "0080B23C001B6606","70FF60000082B23C"
820 DATA "000A6700008CB23C","0020670000084B23C"
830 DATA "00D0660AB7D667D4","558B7C15601 2B23C"
840 DATA "00D666C67C16548B","B7CA6604558B60BC"
850 DATA "7200701161 3A611C","2006613467 0E7018"
860 DATA "3204BC7C00156702","4441612472007011"
870 DATA "611E60827000303 5","E802D08C22407400"
880 DATA "1431E800B4476B02","3407528970074E44"
890 DATA "4E434A806708B07C","FFFC6702588F4A80"
900 DATA "4E75220548457028","61E64841702960E0"
910 DATA "42474847226E0058","7202BE3C00036712"
920 DATA "70003035E802D9C0","D234E80052410881"
930 DATA "00002C013478011A","4E92226E005893C6"
940 DATA "2D49005 8BE3C0003","660C33B5E804E800"
950 DATA "28077 0004E755546","4231E80013B4E800"
960 DATA "E801528C528951CE","FFF460E472001228"
970 DATA "0045484112280046","60D8720032280 1C"
980 DATA "3428002682C25341","4841322800284841"
990 DATA "60C0000000000003","FDB2065F4E414D45"
1000 DATA "2400FDF2055F4445","4624FDE6055F4445"
1010 DATA "4625","*",52617
```

Listing one refers to several labels in the second part, which has been held over till next month because of limited space. CHOOSE labels the routine which allows a name to be selected from the menu. DONE and STACKER return the function result to SuperBasic. QUIT_OUT returns the error code in D0, discarding the last return address with ADDQ.L #4, A7 so the program returns directly to the original caller.

Subroutines

GET_CSIZES and GET_COLOURS are EXTOP subroutines which read values from the Channel Definition and return two each, in register D1. PRINT_NAME is called with A5 holding the offset of a Name Table entry; it prints the name to the channel specified by the ID in A0. TRAP3 performs TRAP #3, returning only ERR_OR or ERR_OK. Other errors are propagated to QUIT_OUT.

Next issue I shall conclude the assembler code and commentary for these functions, and explore the table of Qdos vectors, documenting and explaining uses for vectors that have never before been published. DIY Toolkit continues to extend the QL and compatibles with concise general-purpose routines that make the machine easier to use and program. I welcome suggestions from readers, care of *QL World*.

QL Hardware

Ingenieurbüro
Wilfried Krummrey
Postfach 47 04 41
D-1000 Berlin 47

QL Software

QDesign

Das erstklassige **Grafikprogramm** für den Sinclair QL. The best **graphics program** for the Sinclair QL.
QDesign deutsch oder/or english DM 111,– / £ 37
Für/For HP LaserJet, DeskJet, InkJet: QDesign Laser deutsch oder/or english DM 129,– / £ 43

VecEdit

Der **Vektorfont-Editor** für *QDesign*. The **vector font editor** for *QDesign*.
VecEdit deutsch oder/or english DM 54,– / £ 18

Neu! **SYSTEM** Neu!

Die **Systemerweiterung** und Ergänzung zum Toolkit II. Mehr als 60 neue Befehle. UNIX-feeling auf dem QL.
This **Toolkit** starts where TK2 finished. More than 60 new commands. Unix feeling at the QL.
SYSTEM deutsch oder/or english DM 99,– / £ 33

QL-Tastatur-90

Das erste vollkompatible **QL Tastaturinterface**. The first fully compatible **QL keyboard interface**.
QL-Tastatur-90 deutsch oder/or english DM 149,– / £ 49.70

FOR AUSFÜHRLICHERE INFORMATIONEN: FORDERN SIE BITTE UNSERE KOSTENLOSE PREISLISTE AN!
FOR MORE INFORMATION: PLEASE ORDER OUR FREE OF CHARGE ENGLISH PRICE-LIST!

Alle Preise sind Exportpreise excl. VAT vorbehaltlich Liefermöglichkeit.
Versandkostenpauschale DM 10,– / £ 3.50 bei Vorkasse durch Verrechnungsscheck oder Postanweisung.
Exportprices excl. VAT. E&OE. Please add DM 10,– / £ 3.50 for postage. Payment: Cheques or Postal Orders.

SOFTWARE FILE

PD1

INFORMATION
Program: *Disk PD1*
Price: Call supplier
Supplier: QUBBESoft P/D
38 Brunwin Road
Rayne
Braintree
Essex CM7 5BU.
Tel. (0376)-47852

Qubbesoft is a new name on the public domain market. Bryan Davies looks at their first disk.

The review disk has been sitting, patiently awaiting attention for several months, and there are now additional public domain disks available from QubbeSoft. Specifically, a C language compiler is being offered. PD programs are relatively few and far between for the QL, but low prices are appreciated, so there is the chance for a PD supplier to be modestly successful.

Unusual

An initial look at the contents of disk PD1 suggested we had something a bit unusual here. Several files are of zero size, and a View of the boot file showed it to consist almost entirely of the 'splodge' characters that non-programmers find so mystifying. Nevertheless, it ran and produced a good, clear screen, listing the programs supplied on the disk. A highlight bar can be moved over the programs, and running a program is as simple as pressing ENTER when it is highlighted, with the exception that CTRL-C has to be pressed subsequently to grab the cursor for some EXECable programs. The programs listed were *Imagix diamond edition* V4.30, *The Cataloguer* V1.12, *Touch-it!* V2.15, *The Mandelbrot Machine* V1.69, *Tools* V1.07, *Variable Memory Shrink* V2.50, *Soft Eprom* V1.00, *Super Kit Merger* V1.00, *Qpuzzle* V1.00 and *Coût/Km et Amort* V1.00.

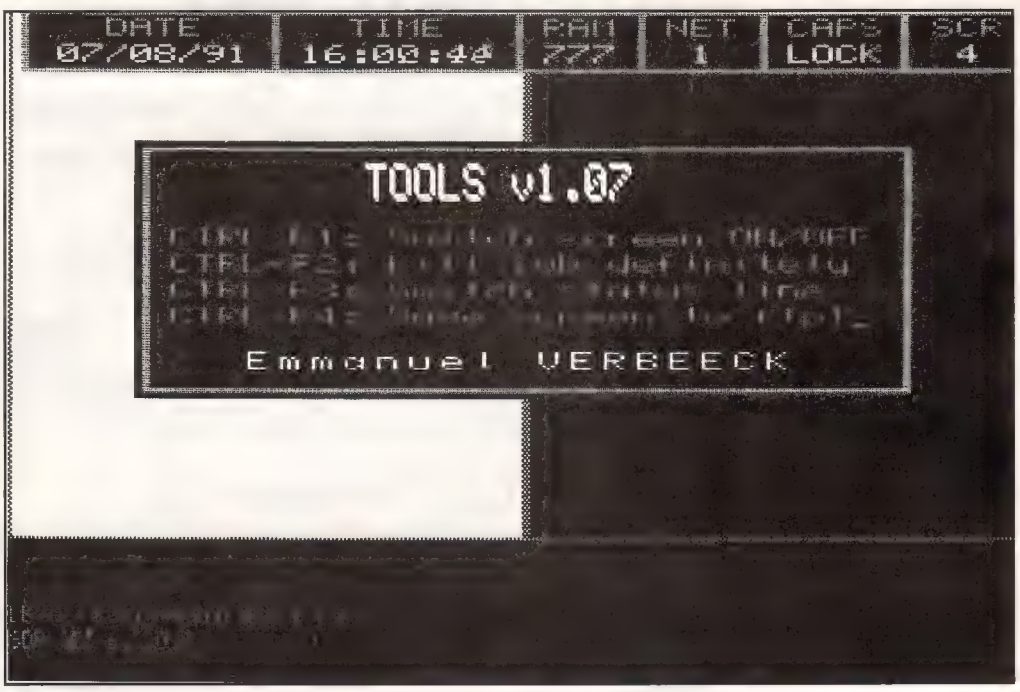
It is quite likely some of these programs have reached later versions by now. One thing it is unreasonable to expect at PD prices is a book of instructions, and one can grind to a temporary halt at the first screen of the first program, wondering not only what to do next but also what the program is for. *Imagix* turned out to be a screen dump utility, with a large selection of parameters. Output quality is good. In view of the uncertainty of what was being entered into, the first print was initiated by pressing ENTER for every selection on the menu; this minimalist approach turned out to be satisfactory for a demonstration of the program's capability.

The dump handled all four basic colours of Mode 4 well, which is something the SDUMP command in some interfaces doesn't seem to do without 'tweaking'. There was little trace of the horizontal white lines which often spoil dumps of largely-black backgrounds. There are enough variables to enable the user to select and size screens to suit any requirement; a moving bar indicates how far the dump has progressed. A neat little program.

The Cataloguer lists the files on disks or microdrive cartridges, to the screen or a printer. It doesn't cater for hard disk. The listing to both screen and printer were good from flp1_ (or fdk1_), but no listings were obtained from the other proffered devices – flp2_, fdk2_, mdv1_ and mdv2_. It would have been useful to be able to write listings to a disk file.

Touch-it is a macro-generation routine. It allows you to write macros for recall with the ALT key plus alphanumeric keys,

Tools – the screen save worked fine, but the on/off switch didn't.



rather in the fashion of the Toolkit ALTKEY function. For a user who does not have an interface with Toolkit functions, this could be a cheap way of obtaining one of the most useful of them. A RESPR/LBYTES/CALL command line is needed, to load the supplied extentions files and make the creation of the macros possible.

Fractals

We see so much mention of fractals and Mandelbrot that it might seem no computer is complete without a program to generate fractal patterns. Unfortunately, one of the features of these patterns is the long time taken by some programs to display them on the QL screen. This doesn't appear to be a particular problem with *The Mandelbrot Machine*, but my knowledge of what fractal patterns should look like is insufficient to judge whether or not those produced by the program are authentic. They look reasonable, though. There were plenty of parameters to vary, and a Help screen.

Tools

The next program on the list is *Tools*, and this displays a menu of four options – 'switch the screen off/on, kill job definitely, switch status line, save screen to flp1_'. The first option worked instantly, making the screen black as soon as CTRL-F1 was keyed. Despite the impression given by the menu text, it did not turn the screen back on again when the same keying was used, leaving the choice of either waiting an uncertain time to see if it came back on again of its own accord, or resetting. After more than five minutes wait, the latter seemed the only course open. The second option did what it said, removing Tools from the task list. The status line appears across the top of the screen, giving the current date and time, the network number, the CAPS LOCK key status, and two items labelled 'RAM' and 'SCR'. Presumably, the former indicates free memory in KB; it showed 777 on my system, which was about right. The latter may be a reference to screen window number, or number of windows. The screen save worked fine (see the illustration of Tools V1.07 screen).

There may not be too many occasions when you wish your system had less than the installed amount of memory, but disk/memory interfaces tend to have a command for reducing the utilised memory to 128 KB, to allow the use of programs which work only with that amount. The *Variable Memory Shrink* routine extends the options to 128/256/384/512/640/768/896 KB. The last value seems a bit odd, being the maximum one could normally get (until the advent of the Gold Card). The function works, although it is a bit disconcerting to find a Trump Card made useless (apart from its memory), neither TK2 nor the floppy disk driver being available. The Trump Card command, RES 128, does at least leave you with disk drives. The 768 KB value proved to be unavailable, causing the program to halt, but all the other values worked.

SoftEprom was not something that could be checked on my system. Its function appears to be to alter the memory address of an eprom cartridge.

Super Kit Merger appeared to be something that would be quite useful – a sort of filter, to get rid of duplicate extensions to the Qdos command set. If you've ever used the Toolkit command EXTRAS · and inspected the list presented on the screen, you are likely to have wondered why some 'names' appear more than once. The duplicates not only take up memory space but they may cause problems if there is more than one apparent response available when a program calls a function. Unfortunately, the small list of extensions files given to Super Kit Merger to digest and reduce to a slimmer, single file just caused it to halt without any explanatory message. A second, smaller list resulted in a 'newly-created toolkit' which was four bytes in size – not likely to add much to Qdos! Two more attempts resulted in one more each of the same results.

Qpuzzle starts up with a very striking, flickering border around a few introductory words (the illustration doesn't do it full justice); the words are

Qpuzzle – good for your French

French except for 'PUZZLE'. As with tv, seeing something on the screen in another language can make it somehow less boring than when it is in one's own language. In fact, this program is perhaps the best of the bunch. It is essentially one of those games where you move blocks on a board to get a certain pattern, but in this case the program does it for you. You simply sit back and watch the squares moving. Leastways, that's how it seemed at first – until the instructions were deciphered. What you have to do is take over from the computer and use the cursor keys and press Space until the pattern is correct. The pattern is a No Entry sign, into which the form of a ghost clambers. Very nicely done.

The last program is entitled *Coût/Km et Amort* V1.00, which may not seem anything to the average English-speaking user, but it is a simple calculator to tell you how much your motoring is costing, on a Francs per kilometre basis, and also the yearly depreciation. There are 18 headings to give information under, so brush up on your French!

Errors

There were the usual problems, such as untrapped (and unexplained) errors, with some of the programs as is usual with commercial items at the lower end of the price scale. Users could learn to live with most of them, but it looked as though more program testing (maybe on more QL versions) is desirable. The lack of written or disk file instructions is, to some extent, balanced by Help screens, but you would need to print these out. The uninitiated user would have little idea of what some of the parameters are for. Screen presentation is good, and it is clear a fair amount of thought has gone into both this and the basic mechanisms of the programs. All of the programs are credited to the same person, incidentally. The disk would provide an interesting few hours, and might give some ideas to those developing their own programs. At normal PD prices, there's little to lose.

CATCH JACK

Jose Carlos de Prada is looking for a villain . . .

Catch Jack is an original game. As you take an evening stroll through Old London Town you will witness a murder and, for a brief moment, you will glimpse the face of the murderer. Reporting the murder to the police, you will be asked to construct an identikit picture of the assassin. Your attempts at remembering important facial details will be crucial to police efforts to apprehend the villain.

The program oozes atmosphere with its short descriptions of each scene accompanied by some clever graphics and spooky music. The listing is neatly laid out and divided into a large number of procedures, mostly devoted to graphics co-ordinates. Type these in very carefully!

To play the game, select a level of difficulty from 1 to 5. Watch the screen as the murder is committed and the murderer's face comes into view. You will then be taken to the police station to build up the identikit. Select each facial feature from a menu using the up, down and Return keys. The program will then award points according to how close you came to describing Jack. Only if your description is perfect will you have the pleasure of seeing Jack behind bars.

```
100 MODE 8
110 RANDOMISE
120 inic
130 REPeat game
140   menu_levels
150   crime
160   interrogate
170   score
180   IF another : WINDOW 512,216,0,0 : CLS : ELSE : EXIT game
190 END REPeat game
200 WINDOW 256,202,256,0 : BORDER 1,255
210 WINDOW #2,256,202,0,0 : BORDER #2,1,255 : CLOSE #3 : MODE 4
220 REMark ==========================================
230 DEFine PROCedure jaw_1
240   LINE #3,20,40 TO 35,10 : ARC #3 TO 55,10,PI/1.5 : LINE #3 TO 70,40
250 END DEFine
260 REMark ==========================================
270 DEFine PROCedure jaw_2
280   LINE #3,20,40 TO 22,20 TO 40,10 TO 50,10 TO 68,20 TO 70,40
300   LINE #3,45,10 TO 45,13
310 END DEFine
320 REMark ==========================================
330 DEFine PROCedure jaw_3
340   LINE #3,20,40 TO 22,20 : ARC #3 TO 68,20,PI/1.3 : LINE #3 TO 70,40
350   ARC #3,37,10 TO 53,10,PI#2/4
360 END DEFine
370 REMark ==========================================
380 DEFine PROCedure jaw_4
390   FILL #3,1 : LINE #3,25,20 TO 25,15 TO 45,3 TO 35,16
400   ARC #3 TO 25,20,-PI/3 : FILL #3,0 : FILL #3,1
410   LINE #3,45,3 TO 65,15 TO 65,20 : ARC #3 TO 55,15,-PI/3
420   LINE #3 TO 45,3 : FILL #3,0 : FILL #3,1
430   ARC #3,35,15 TO 45,20,PI/3 TO 55,15,PI/3 : LINE #3 TO 45,3 TO 35,15
440   FILL#3,0
450   LINE #3,25,20 TO 20,40
460   LINE #3,65,20 TO 70,40
470 END DEFine
480 REMark ==========================================
490 DEFine PROCedure jaw_5
500   FILL #3,1 : LINE #3,20,40 TO 20,10 TO 25,20 TO 20,40 : FILL #3,0
510   FILL#3,1 : ARC #3,35,13 TO 25,20,-PI/2 : LINE #3 TO 20,10 TO 35,13
520   FILL #3,0
530   FILL #3,1 : LINE #3,65,20 TO 70,10 TO 70,40 TO 65,20 : FILL #3,0
540   FILL #3,1 : ARC #3 TO 55,13,-PI/2 : LINE #3 TO 70,10 TO 65,20
550   FILL #3,0
560   FILL #3,1 : LINE #3,35,13 TO 20,10 : ARC #3,20,10 TO 70,10,PI/2
570   LINE #3 TO 55,13 : ARC #3 TO 45,20,-PI/3 TO 35,13,-PI/3 : FILL #3,0
580 END DEFine
590 REMark ==========================================
600 DEFine PROCedure hair_1
610   LINE #3,20,40 TO 20,75 : ARC #3 TO 70,75,-PI/1.5
620   LINE #3 TO 70,40 TO 67,40 TO 67,50 TO 65,60 TO 65,75
630   ARC #3 TO 25,75,PI/4 : LINE #3 TO 25,60 TO 23,50 TO 23,40 TO 20,40
640   ARC #3,20,40 TO 20,52,-PI/2
650   ARC #3,70,40 TO 70,52,PI/2
660 END DEFine
670 REMark ==========================================
680 DEFine PROCedure hair_2
690   FILL #3,1 : LINE #3,20,40 TO 20,76 : ARC #3,20,75 TO 45,90,-PI/3
1390 REMark ==========================================
1400 DEFine PROCedure eyes_5
1410   FILL #3,1 : LINE #3,27,60 TO 40,60 : ARC #3 TO 42,58,-PI/2
1420   ARC#3 TO 27,50,-PI/2 TO 25,52,PI/2 TO 25,58,PI/3 TO 27,60,PI/2
1430   FILL #3,0
1440   FILL #3,1 : LINE #3,63,60 TO 50,60 : ARC#3 TO 48,58,PI/2 TO 63,50,PI/2
1450   ARC#3,63,49 TO 65,52,-PI/3 TO 65,58,-PI/3 TO 63,60,-PI/2 : FILL #3,0
1460   LINE #3,39,60 TO 50,60 : LINE #3,42,58 TO 48,58
1470 END DEFine
1480 REMark ==========================================
1490 DEFine PROCedure mouth_1
1500   ARC #3,30,24 TO 60,24,PI/4 TO 30,24,PI/4 : LINE #3 TO 60,24
1510 END DEFine
1520 REMark ==========================================
1530 DEFine PROCedure mouth_2
1540   ARC #3,30,24 TO 60,24,PI/2.5 TO 55,27,PI/3 TO 30,24,PI/3
1550   LINE #3 TO 60,24
1560 END DEFine
1570 REMark ==========================================
1580 DEFine PROCedure mouth_3
1590   ARC #3,38,24 TO 53,24,PI/3 TO 38,24,PI/3 : LINE #3 TO 53,24
1600 END DEFine
1610 REMark ==========================================
1620 DEFine PROCedure mouth_4
1630   ARC #3,38,24 TO 52,24,PI/1.5 TO 45,26,PI/2 TO 38,24,PI/2
1640   LINE #3 TO 52,24
1650 END DEFine
1660 REMark ==========================================
1670 DEFine PROCedure mouth_5
1680   ARC #3,36,24 TO 54,24,PI/2 TO 36,24,PI/3 : LINE #3 TO 54,24
1690   FILL #3,1 : LINE #3,30,25 TO 45,28 TO 45,30 TO 30,27 TO 30,25
1700   FILL #3,0 : FILL #3,1
1710   LINE #3,45,28 TO 60,25 TO 60,27 TO 45,30 TO 45,28 : FILL #3,0
1720 END DEFine
1730 REMark ==========================================
1740 DEFine PROCedure nose_1
1750   LINE #3,43,50 TO 43,35 TO 45,30  TO 47,35 TO 47,50
1760   LINE #3,43,38 TO 40,35 TO 43,32 : LINE #3,47,38 TO 50,35 TO 47,32
1770 END DEFine
1780 REMark ==========================================
1790 DEFine PROCedure nose_2
1800   ARC #3,40,35 TO 50,35,-PI : ARC #3,35,35 TO 41,39,-PI/2
1810   ARC #3,55,35 TO 49,39,PI/2 : LINE #3,42,35 TO 44,35
1820   LINE #3,48,35 TO 46,35
1830 END DEFine
1840 REMark ==========================================
1850 DEFine PROCedure nose_3
1860   ARC #3,42,50 TO 43,39,PI/2 TO 45,30,PI/1.5 TO 47,39,PI/1.5 TO 48,50,PI/2
1870   ARC #3,43,39 TO 43,32,PI
1880   ARC #3,47,39 TO 47,32,-PI
1890 END DEFine
```

```
700   LINE #3 TO 45,80 : ARC #3 TO 25,75,PI/4
710   LINE #3 TO 25,60 TO 23,50 TO 23,40 TO 20,40 : FILL #3,0
720   FILL #3,1 : LINE #3,65,75 TO 65,60 TO 67,50 TO 67,40 TO 70,40 TO 70,75
730   ARC #3 TO 45,90,PI/3 : LINE #3 TO 45,80 : ARC #3 TO 65,75,-PI/4
740   FILL #3,0
750   ARC #3,20,40 TO 20,52,-PI/2
760   ARC #3,70,40 TO 69,52,PI/2
770 END DEFine
780 REMark ================================================================
790 DEFine PROCedure hair_3
800   ARC #3,20,40 TO 15,55,-PI/3 : ARC #3 TO 10,70,-PI/2
810   ARC #3 TO 15,85,-PI : ARC #3 TO 25,90,-PI
820   ARC #3 TO 35,95,-PI : ARC #3 TO 45,95,-PI
830   ARC #3 TO 55,95,-PI : ARC #3 TO 65,90,-PI
840   ARC #3 TO 75,80,-PI : ARC #3 TO 75,60,-PI/2
850   ARC #3 TO 70,40,-PI/3
860   LINE #3,70,40 TO 72,60 : ARC #3 TO 60,75,-PI/3
870   ARC #3 TO 50,80,-PI/2 : ARC #3 TO 40,80,-PI/2
880   ARC #3 TO 25,70,-PI/2 : ARC #3 TO 18,60,-PI/2
890   LINE #3 TO 20,40
900 END DEFine
910 REMark ================================================================
920 DEFine PROCedure hair_4
930   FILL #3,1
940   ARC #3,20,40 TO 15,55,-PI/3 TO 10,70,-PI/2 TO 15,85,-PI TO 25,89,-PI/1.5
950   LINE #3,25,90 TO 40,80 : ARC #3,40,81 TO 25,70,-PI/2 TO 18,60,-PI/2
960   LINE #3 TO 20,40 : FILL #3,0
970   FILL #3,1 : LINE #3,25,90 TO 40,80 : ARC #3 TO 51,80,PI/2
980   LINE #3 TO 56,93 : ARC #3 TO 25,90,PI/2 : FILL #3,0
990   FILL #3,1 : LINE #3,55,93 TO 50,80
1000   ARC #3 TO 60,75,PI/1.5 TO 72,60,PI/1.5 : LINE #3 TO 70,40
1010   ARC #3 TO 75,60,PI/3 TO 75,80,PI/1.5 TO 65,90,PI/1.5 TO 55,93,PI
1020   FILL #3,0
1030 END DEFine
1040 REMark ================================================================
1050 DEFine PROCedure hair_5
1060   LINE #3,20,40 TO 15,60 : ARC #3 TO 75,60,-PI : LINE #3 TO 70,40
1070   ARC #3,20,40 TO 15,60,-PI/2
1080   ARC #3,70,40 TO 75,60,PI/2
1090 END DEFine
1100 REMark ================================================================
1110 DEFine PROCedure eyes_1
1120   ARC #3,30,55 TO 40,55,PI/2 TO 30,55,PI/2 : CIRCLE #3,35,55,2
1130   POINT #3,35,55 : ARC #3,40,60 TO 30,60,PI/2
1140   ARC #3,50,55 TO 60,55,PI/2 TO 50,55,PI/2 : CIRCLE #3,55,55,2
1150   POINT #3,55,55 : ARC #3,60,60 TO 50,60,PI/2
1160 END DEFine
1170 REMark ================================================================
1180 DEFine PROCedure eyes_2
1190   ARC #3,30,55 TO 40,55,PI/2 TO 30,55,PI/2 : FILL #3,1
1200   CIRCLE #3,35,55,2 : FILL #3,0 : ARC #3,40,60 TO 30,60,PI/2
1210   ARC #3,50,55 TO 60,55,PI/2 TO 50,55,PI/2 : FILL #3,1
1220   CIRCLE #3,55,55,2 : FILL #3,0 : ARC #3,60,60 TO 50,60,PI/2
1230 END DEFine
1240 REMark ================================================================
1250 DEFine PROCedure eyes_3
1260   ARC #3,30,55 TO 40,55,PI/3 TO 30,55,PI/3 : FILL #3,1
1270   CIRCLE #3,35,55,1 : FILL #3,0 : ARC #3,40,60 TO 30,60,PI/3
1280   ARC #3,50,55 TO 60,55,PI/3 TO 50,55,PI/3 : FILL #3,1
1290   CIRCLE #3,55,55,1 : FILL #3,0 : ARC #3,60,60 TO 50,60,PI/3
1300 END DEFine
1310 REMark ================================================================
1320 DEFine PROCedure eyes_4
1330   ARC #3,30,55 TO 40,55,PI/2 TO 30,55,PI/2 : CIRCLE #3,35,55,2
1340   POINT #3,35,55
1350   ARC #3,50,55 TO 60,55,PI/2 TO 50,55,PI/2 : CIRCLE #3,55,55,2
1360   POINT #3,55,55
1370   CIRCLE #3,35,55,8 : CIRCLE #3,55,55,8 : LINE #3,43,55 TO 47,55
1380 END DEFine
1900 REMark ================================================================
1910 DEFine PROCedure nose_4
1920   LINE #3,42,50 TO 40,35 : ARC #3 TO 50,35,PI : LINE #3 TO 48,50
1930   ARC #3,40,40 TO 42,31,PI
1940   ARC #3,50,40 TO 48,31,-PI
1950 END DEFine
1960 REMark ================================================================
1970 DEFine PROCedure nose_5
1980   LINE #3,43,50 TO 40,35 : LINE #3,47,50 TO 50,35 : CIRCLE #3,45,35,4
1990   ARC #3,40,38 TO 41,32,PI
2000   ARC #3,50,38 TO 49,32,-PI
2010 END DEFine
2020 REMark ================================================================
2030 DEFine PROCedure portrait
2040   jaw_5
2050   hair_1
2060   eyes_4
2070   mouth_5
2080   nose_3
2090 END DEFine
2100 REMark ================================================================
2110 DEFine PROCedure menu_levels
2120   WINDOW 512,256,0,0
2130   PAPER 0 : INK 7
2140   CLS
2150   REPeat choice
2160     AT 18,5 : PRINT  "Press a level number  (1 to 5)"
2170     k=CODE(INKEY$(-1))
2180     IF k>48 AND k<54 : k=k-48 : EXIT choice : ELSE BEEP 500,50
2200   END REPeat choice
2210   D=(5-k)*50
2220   CLS : AT 5,10 : PRINT "LONDON, AUTUMN  1888" : AT 16,4
2230   PRINT"While  you are  taking  a walk  in the"!"night,  a murderer  goes r
ound through the"!"streets protected by the mist."!"   In the  distance  you can
  see a feeble"!"light ..."
2240   PAUSE 300 : CLS
2250 END DEFine
2260 REMark ================================================================
2270 DEFine PROCedure crime
2280   scene
2290   BEEP 26700,4,2,32,-1,5,3,1
2300   PAUSE 100
2310   BEEP 4000,255,150,2,-3,2,14,2
2320   victim : PAUSE 50
2330   FOR i=1 TO 4
2340     BEEP 1000,20,0,0,0,0,0,10
2350     PAUSE 10
2360     BEEP 1000,30,30,0,0,0,0,10
2370     PAUSE 20
2380   END FOR i
2390   WINDOW #3,512,256,0,0 : INK #3,7
2400   FILL #3,1
2410   LINE #3,0,100 TO 0,0 TO 190,0 TO 80,100
2420   FILL #3,0
2430   PAPER #3,7 : INK #3,0
2440   knife
2450   murderer
2460   PAUSE D
2470   CLS
2480   l=30
2490   FOR i=1 TO 10
2500     l=l-5
2510     IF l<5 : l=5
2520     BEEP 1000,30,30,0,0,0,0,10 : PAUSE l*1.5 : BEEP 1000,30,30,0,0,0,0,10
2530     PAUSE l
2540   END FOR i
2550 END DEFine
2560 REMark ================================================================
2570 DEFine PROCedure murderer
2580   m=RND (1 TO 5) : p=RND (1 TO 5) : o=RND (1 TO 5) : b=RND (1 TO 5)
```

```
2590    n=RND (1 TO 5)
2600    face
2610 END DEFine
2620 REMark ================================================================
2630 DEFine PROCedure face
2640    s_jaw(m)
2650    s_hair(p)
2660    s_eyes(o)
2670    s_mouth(b)
2680    s_nose(n)
2690 END DEFine
2700 REMark ================================================================
2710 DEFine PROCedure interrogate
2720    WINDOW 256,256,0,0
2730    PAPER 2 : INK 7 : BORDER 2,255 : CLS
2740    WINDOW #3,256,256,256,0 : PAPER #3,7 : BORDER #3,2,255 : INK #3,0
2750    CLS#3
2760    SCALE #3,120,0,0
2770    CSIZE 3,1
2780    AT 3,5 : PRINT "STATION"
2790    FLASH 1
2800    AT 1,5 : PRINT "POLICE"
2810    FLASH 0
2820    CSIZE 2,0
2830    AT 11,3
2840    PRINT "You are  the only"!" witness of a crime."!"   Therefore you are"!"
 the  only  one  who"!" can  identify   the"!" murderer."!"   Select  the right"
!" features  to  build"!" a portrait."
2850    PAUSE 500
2860    w=0
2870    RESTORE 5140
2880    FOR i=1 TO 5
2890      CLS
2900      CSIZE 3,1
2910      READ title$
2920      AT 2,5 : PRINT title$
2930      CSIZE 2,0 : u=1 : OVER -1 : OVER#3,-1 : INK#3,7
2940      AT 7,8 : PRINT "1-"
2950      AT 8,8 : PRINT "2-"
2960      AT 9,8 : PRINT "3-"
2970      AT 10,8 : PRINT "4-"
2980      AT 11,8 : PRINT "5-"
2990      AT 16,1 : INPUT "↑ ↓     to view"!" ENTER   to choose"
3000      REPeat sel_features
3010        SELect ON i
3020          =1 : s_jaw(u)
3030          =2 : s_hair(u)
3040          =3 : s_eyes(u)
3050          =4 : s_mouth(u)
3060          =5 : s_nose(u)
3070        END SELect
3080        BLOCK 24,10,96,(6+u)*10,4 : vu=u
3090        ch=CODE(INKEY$(-1))
3100        SELect ON ch
3120          =216 : IF u<5 : u=u+1 : ELSE : BEEP 200,20
3140          =208 : IF u>1 : u=u-1 : ELSE : BEEP 200,20
3150          =10  : EXIT sel_features
3160          =REMAINDER : BEEP 200,20
3170        END SELect
3180        BLOCK 24,10,96,(6+vu)*10,4
3190        SELect ON i
3200          =1 : s_jaw(vu)
3210          =2 : s_hair(vu)
3220          =3 : s_eyes(vu)
3230          =4 : s_mouth(vu)
3240          =5 : s_nose(vu)
3250        END SELect
3260      END REPeat sel_features
3270      OVER 1 : INK#3,0 : OVER#3,1
3920    AT 2,5 : PRINT "====="
3930    CSIZE 2,0
3940    AT 6,4 : PRINT "RIGHT"!"     FEATURES: ";w
3950    CSIZE 3,1
3960    IF w=5
3970      FLASH 1
3980      AT 8,3 : PRINT "CAPTURED"
3990      FLASH 0
4000      h=14
4010      FOR i=1 TO 8
4020        BLOCK #3,10,250,h,0,0
4030        h=h+30
4040      END FOR i
4050      FOR j=1 TO 10
4060        BEEP 10000,33
4070        PAUSE 30
4080        BEEP 10000,22
4090        PAUSE 30
4100      END FOR j
4110    ELSE
4120      WINDOW#3,120,120,60,90 : BORDER#3,2,255 : CLS#3 : face
4130    END IF
4140    IF w<5 AND w>0 : melo
4150    IF w=0 : BEEP 30000,250 : PAUSE 200
4160 END DEFine
4170 REMark ================================================================
4180 DEFine FuNction another
4190    CSIZE 2,0 : otr%=-1
4200    AT 21,3 : PRINT "Another game?            (Y/N)"
4210    REPeat question
4220      z$=INKEY$(-1)
4230      IF z$="Y" OR z$="y" OR z$="N" OR z$="n"
4240        EXIT question : ELSE : BEEP 500,50
4250      END IF
4260    END REPeat question
4270    IF z$= "Y" OR z$= "y" : RETurn -1
4280    IF z$= "N" OR z$= "n" : RETurn 0
4290 END DEFine another
4300 REMark ================================================================
4310 DEFine PROCedure melo
4320    LOCal D,T,p
4330    FOR i=1 TO 2
4340      RESTORE 4400
4350      FOR j=1 TO 13
4360        READ D,T,p
4370        BEEP D,T : PAUSE p
4380      END FOR j
4390    END FOR i
4400 DATA 16000,47,25
4410 DATA 32700,33,50
4420 DATA 32700,26,50
4430 DATA 32700,28,17
4440 DATA 8000,33,10
4450 DATA 8000,35,10
4460 DATA 8000,38,10
4470 DATA 8000,41,10
4480 DATA 8000,44,6.5
4490 DATA 8000,44,10
4500 DATA 8000,47,30
4510 DATA 8000,47,25
4520 DATA 4000,33,40
4530 END DEFine
4540 REMark ================================================================
4550 DEFine PROCedure knife
4560    LINE #3,85,0 TO 95,10 TO 107,10
4570    ARC #3 TO 107,7,-PI TO 107,3,-PI TO 107,0,-PI
4580    ARC #3,110,9 TO 110,5,-PI
4590    LINE #3,93,10 TO 107,10 TO 107,12 TO 93,12 TO 93,10
4600    LINE #3,96,12 TO 96,60 : LINE #3,104,12 TO 104,60
```

```
3280      SELect ON i
3290        =1 : s_jaw(u)
3300             IF u=m : w=w+1
3310        =2 : s_hair(u)
3320             IF u=p : w=w+1
3330        =3 : s_eyes(u)
3340             IF u=o : w=w+1
3350        =4 : s_mouth(u)
3360             IF u=b : w=w+1
3370        =5 : s_nose(u)
3380             IF u=n : w=w+1
3390      END SELect
3400    END FOR i
3410 END DEFine
3420 REMark ======================================================================
3430 DEFine PROCedure s_jaw (x)
3440   SELect ON x
3450      =1 : jaw_1
3460      =2 : jaw_2
3470      =3 : jaw_3
3480      =4 : jaw_4
3490      =5 : jaw_5
3495    END SELect
3500 END DEFine
3510 REMark ======================================================================
3520 DEFine PROCedure s_hair (x)
3530    SELect ON x
3540      =1 : hair_1
3550      =2 : hair_2
3560      =3 : hair_3
3570      =4 : hair_4
3580      =5 : hair_5
3585    END SELect
3590 END DEFine
3600 REMark ======================================================================
3610 DEFine PROCedure s_eyes (x)
3620    SELect ON x
3630      =1 : eyes_1
3640      =2 : eyes_2
3650      =3 : eyes_3
3660      =4 : eyes_4
3670      =5 : eyes_5
3675    END SELect
3680 END DEFine
3690 REMark ======================================================================
3700 DEFine PROCedure s_mouth (x)
3710    SELect ON x
3720      =1 : mouth_1
3730      =2 : mouth_2
3740      =3 : mouth_3
3750      =4 : mouth_4
3760      =5 : mouth_5
3765    END SELect
3770 END DEFine
3780 REMark ======================================================================
3790 DEFine PROCedure s_nose (x)
3800    SELect ON x
3810      =1 : nose_1
3820      =2 : nose_2
3830      =3 : nose_3
3840      =4 : nose_4
3850      =5 : nose_5
3855    END SELect
3860 END DEFine
3870 REMark ======================================================================
3880 DEFine PROCedure score
3890    CLS
3900    CSIZE 3,1
3910    AT 1,5 : PRINT "SCORE"
4610    LINE #3,100,12 TO 100,80 : INK #3,2 : FILL #3,1
4620    LINE #3,104,60 TO 96,45 TO 96,60
4630    ARC #3,96,59 TO 100,80,-PI/5 TO 104,59,-PI/5 : LINE #3,96,50 TO 94,42
4640    ARC #3 TO 96,41,PI : LINE #3 TO 96,45 : INK #3,0 : FILL #3,0
4650 END DEFine
4660 REMark ======================================================================
4670 DEFine PROCedure scene
4680    FILL 1 : INK 6,0 : CIRCLE 125,85,25 : FILL 0
4690    INK 6,2
4700    FILL 1 : CIRCLE 125,85,20 : FILL 0
4710    INK 6
4720    FILL 1 : CIRCLE 125,85,15 : FILL 0
4730    INK 1
4740    FILL 1 : LINE 125,95 TO 130,90 TO 120,90 TO 125,95 : FILL 0
4750    FILL 0 : LINE 120,90 TO 122,80 TO 128,80 TO 130,90
4760    LINE 122,80 TO 123,77 TO 127,77 TO 128,80
4770    FILL 1
4780    LINE 124,77 TO 121,13 TO 119,13 TO 119,10 TO 131,10 TO 131,13 TO 129,13 T
O 126,77
4790    FILL 0
4800    LINE 0,5 TO 210,5 TO 210,10 TO 70,10 TO 70,5
4810    INK 7
4820 END DEFine
4830 REMark ======================================================================
4840 DEFine PROCedure victim
4850    INK 2
4860    CIRCLE 80,14,4 : LINE 84,15 TO 90,15 : ARC TO 92,13,PI/2
4870    LINE TO 110,12 TO 113,14 TO 113,12 TO 119,15 TO 115,10
4880    LINE 76,15 TO 70,15 : ARC TO 68,13,-PI/2
4890    LINE TO 62,5 TO 60,9 TO 62,8 TO 63,9 TO 64,9 : LINE 70,10 TO 65,5
4900    RESTORE 4960
4910    REPeat blood
4920      READ xa,y,xb
4930      LINE xa,y TO xb,y
4940      IF y=0 : EXIT blood
4950    END REPeat blood
4960 DATA 83,10,85,83,9,85,83,8,85,83,7,85,83,6,85,81,5,90
4970 DATA 80,4,92,78,3,98,80,2,100,81,1,98,84,0,90
4980 END DEFine
4990 REMark ======================================================================
5000 DEFine PROCedure inic
5010    WINDOW 512,256,0,0
5020    PAPER 0 : CLS
5030    AT 2,16 : PRINT "=========="
5040    AT 3,16 : PRINT "CATCH JACK"
5050    AT 4,16 : PRINT "=========="
5060    AT 6,9 : PRINT "by Jose Carlos de Prada"
5070    AT 8,14 : PRINT "December 1987"
5080    OPEN #3,scr_200x150a150x100
5090    PAPER #3,7 : INK #3,2 : BORDER #3,3,2 : CLS #3
5100    portrait
5110    melo
5120 END DEFine
5130 REMark ======================================================================
5140 DATA "JAW","HAIR","EYES","MOUTH","NOSE"
```

aCcLcesBs

FRANCE

QL Contact France. Contact: Jean-Louis Dianoux, 38-40 Rue Stephenson, 75018 Paris, France.

BELGIUM

Club Sinclair BruQsL (Belgium) Contact: Jaques Tasset, Aarlenstraat 104, 1040 Brussels, Belgium

SWEDEN

International QL Conference bulletin board system (Swedish and English). Contact: Michael Cronsten, System Operator, Jamten-TCL, S Soere 1073, 83030 Lit, Sweden.

USA

New England Sinclair QL User Group (USA) Membership Secretary: Sherm Waterman, PO Box 8763, Boston, MA 02114 8763, USA. Magazine: *NESQLUG News*. Editor: Peter Hale, 195 Central Ave., Chelsea, MA 02150, USA.

NORWAY

Norwegian All Sinclair Association (NASA) Contact: P Monstad, NASA, N-5580 Oelen, Norway. Magazine: *Sinclair Magazine*.

ITALY

Qitaly Club Chairman: Roberto Orlandi, Via Brescia 26, 25039 Travagliato (BS), Italy. Tel. (local) +39 30 6863311. Magazine: *Qitaly Magazine*. Editor: Dr Eros Forenzi, Via Valeriana 44, 23010 Berbenno (SO), Italy. Tel. (local) +39 342 492323.

TURKEY

QL Qlub (Turkey). Contact: Bulent Artuz, Prof. Sitesi B/1 D/5, Etiler 80600, Istanbul, Turkey.

AUSTRIA

DER Computer Club. Contact: Peter Postl, Stiegerg 5, 1150 Wien, Autstria.

ENGLAND

Quanta (UK) Membership Secretary, Bill Newell, 213 Manor Road, Benfleet, Essex SS7 4JD. Magazine: *Quanta*. Editor: Bill Fuggle, 20 Widnes Avenue, Selly Oak, Birmingham B29 6QE.

Bristol sub-group: Roy Brereton, 94 Teignmouth Rd, Clevedon, Avon BS21 6DR.

Essex sub-group: Dave Walker, 22 Kempton's Mead, Potters Bar, Herts EM6 3HZ.

London sub-group: Jeremy Davis, 6 Elmcroft Crescent, Harrow, Middlesex HA2 6HN.

Northern Ireland: Billy Turkington, Fairyhill, Rostrevor, Newry, Co., Down BT34 3BB.

Qubbesoft PD library. Contact: Ron Dunnett, 38 Brunwin Rd, Rayne, Braintree, Essex CM7 5BU.

SCOTLAND

Scottish QL Users Group Contact: Alan Pemberton, 65 Lingerwood Rd., Newtongrange, Midlothian EH22 4QQ. Newsletter.

HOLLAND

Sin_QL_Air (Netherlands) Membership Secretary: Bob Visser, Snelrewaard 6, 2904 SN Capelle, a/d Ijssel, Netherlands. Magazine: *Quasar*. Editor: C H M Biemans, Elzenstraat 5, 5461 CL Veghel, Netherlands.

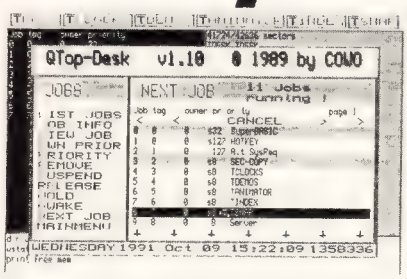
GERMANY

Sinclair QL User Club eV (Germany) Foreign Contact: Franz Herrmann, Talstrasse 21, d-W5460 Ochenfels, West Germany. Magazine: *Quasar*.

SPAIN

Qliper Editor: Marcos Cruz, Acacias 44, 28023 Madrid, Spain. Magazine: *Qliper*.

QTop by COWO ELECTRONIC



Special Christmas Offer!

QTop is the definitive User-Front-End package for all QL compatible computers, the **SINCLAIR QL** itself, all **CST Thor** computers including the powerful **Thor XVI**, the **ATARI ST** and **AMIGA** with QL-Emulators. QTop is the QL's equivalent to ***WINDOWS*** on MS-DOS PC's. The powerful QJUMP extended Pointer Environment is included and if it's installed its facilities are supported. The mainmenus of *QTop* are **DESK**, **DEVICES**, **FILES**, **JOBS**, **KEYBOARD**, **PROGRAMS**, **MY PROGS**, **TOOLS** and **OPTIONS** with over 110 sub-menus. *QTop* is ideal for Gold-Card or Harddisk users as it supports real Sub-Directories. The *QTop* package includes five desktop accessories: **CLOCKS**, **DEMOS**, **ANIMATOR**, **INDEX** and **SNAP**. Software with 41 page manual is available in English or German. Free updates and user support. Now new, totally revised and much more powerful version 1.10. Ask for free product information sheet. QTop is supplied on one 3.5" or 5.25" disc and costs **SFR 70** or **£29.50** including P&P. Send your orders with cheques to:

COWO Electronic

Urs König, Münsterstrasse 4, CH-6210 Sursee, Tel 045-211478 or contact your local distributor: TK Computerware in the UK and QLympic Computer Systems in Germany

IF AN ADVERT IS WRONG, WHO PUTS IT RIGHT?

We do.

The Advertising Standards Authority ensures advertisements meet with the strict Code of Advertising Practice. So if you question an advertiser, they have to answer to us.

To find out more about the ASA, please write to Advertising Standards Authority, Dept. X, Brook House, Torrington Place, London WC1E 7HN.

ASA

This space is donated in the interests of high standards in advertisements.

CARE ELECTRONICS

QPAC II by Tony Tebby

MAKES YOUR QL TRULY MULTI - TASKING

QPAC II Main menu windows adjust automatically to size. Files, directory, view, print, delete, back_p. jobs, pick, Rjob, sort, channels, things exec, wake buttons, Hotkey, Hotjobs. Fully multitasking, allows many programs to run at once. Requires min of 256k expanded memory ..... **SPECIAL OFFER ONLY £39.48b**

Upgrade QRAM to QPAC II return master ..... @29.61b

PLEASE SEND SAE FOR A COPY OF THE QPAC II REVIEW

QPOWER REGULATOR

Switch mode power supply. QL runs cool, stops lock ups (due to voltage drop). Helps filter noise out. Simple to fit (no soldering) £25.85c

QL SUPERTOOL KIT II by Tony Tebby

Over 118 Commands: Full Screen Editor, Key Define Print Using, Last Line Recal, Altkey, Job Control, File Handling, Default Directories, Extended Network

16k Eprom Cartridge Version ..... @£23.50d

Configurable Version of Microdrive ..... @£23.50d

ZITASOFT SOFTWARE by Steve Jones

LOCKSMITHE copies M/DRIVE M DRIVE ..... @£14.10c

4MATTER - LOCKSMITHE copies M/DRIVE - DISC ..... @£23.50c

SIDEWINDER - High resolution printer driver prints full screens or parts of screens from postage stamp size to large banners. Prints sideways, invert, scale, mirror, text insertion ..... @£20.21c

SIDEWINDER PLUS - (for expanded QL's) includes all the features of above. PLUS multiple label printing, desktop publishing files and printer drivers for The LC10 JX80 24 pin and colour printers (Please state 3.5" disc or M Drive) ..... @£23.97c

Upgrade to Sidewinder Plus @£8.46c

SOFTWARE 87 (State MDV or Disc)

TEXT87 (Budget Version) ..... @ £34.98a

FOUNTED 89 ..... @ £15.00

FOUNTED 88 ..... @ £25.00c

TEXT 87/FOUNTED 89/FOUNTEXT 88 ..... @ £94.94b

248B PRINTER DRIVER ..... @ £15.00c

Upgrade to Text 87 V3.00. Return old copy together with ..... @ £25.00d

Send SAE for further info on above.

MIRACLE SYSTEMS PRODUCTS

0K Disc Interface (inc Tool Kit II) ..... @ £99.64b

QL Centronics Printer interface ..... @ £29.61a

QL HARDWARE

QLKeyboard Membrane ..... @ £11.75d

Care Eprom Cartridges each ..... @ £6.11c

ULA CHip ZX8301 ..... @ £15.98c

READY MADE LEADS

RGB QL TO PHONO £6.11c

RGB 8-6 PIN £7.52c

RGB 8-7 PIN HITACHI £7.52c

RGB 8-7 PIN FERGUSON £7.52c

RGB 8 PIN TO SCART £11.75c

RGB 8PIN TO CM8833 MK II £11.75c

6WAY PCC TO 25WAY D £9.87c

TONY TEBBY SOFTWARE (QJUMP)

QLFP (Micro/P disk interface upgrade ..... @ £15.51a

QTYP Type/Spell Checker ..... @ £30.55a

QPAC 1-Desk top accessories, calendar, alarm, calculator, typewriter, digital clock sysmon ..... @ £22.56a

CARE ELECTRONICS
DEPT QL, 15 HOLLAND GDNS
GARSTON, WATFORD,
HERTS, WD2 6JN.
Tel: 0923-894064
Fax: 0923-672102

Pease add carriage
a-£11.75 b=£3.76 c=no charge d=£2.35

HOW TO ORDER

By Post. Enclose your Cheque/PO made payable to CARE Electronics or use ACCESS VISA. Allow 7 days for delivery

☆ sinclair QL ☆

E.E.C. PROVIDE

A COMPLETE COMPUTER SYSTEM FOR £125!

The expandable system for businesses, beginners and experts.
COMPLETE QL's - JS ROM - TESTED PRIOR TO DELIVERY - 6 MONTHS GUARANTEE

Package includes PSU, TV Lead, QL Software 2.35: Quill - Wordprocessor, Abacus - Spreadsheet Archive - Database & Easel - Business Graphics. Comprehensive instruction manual. One years free membership to QUANTA - the independant user group who provide help, newsletters & a library of 400+ programs - mostly free.

Backup units are available which comprise just the QL computer and power supply:
Backup JM QL - £85 -also, enquire about our unique exchange scheme. Computers accepted include: QL, Spectrum - 16k, 48k, 128k, +2, +3, Commodore, BBC etc

UNIVERSAL DISK DRIVES (all our drives are brand new product)

Universal 3.5" 1Mb (unformatted) single - £70, double - £130
Universal 3.5" 2Mb (unformatted) single - £90, double - £170
THE UNIVERSAL DRIVE MAY BE FITTED TO MANY TYPES OF COMPUTER BY ATTACHING A CABLE AND SELECTING THE CORRECT DIP SWITCH SETTINGS. I.E. QL, SPECTRUM, AMIGA, PC)
Uncased 3.5" 1Mb (unformatted) - £39 each
Uncased 3.5" 2Mb (unformatted) - £59 each

QL to PC KEYBOARD ENHANCEMENTS

QL/PC 102 keyboards - £25 UK version, XT/AT switchable - £35
QL/PC keyboard interface - £75 External fitting box & cable - £17

PRINTERS

Seikosha SP1900+ 9 pin dot matrix, up to 192 cps, FX850 emulation - £120
Canon BJ10ex bubblejet, very high quality (360dpi), A4 size, - £259
We can supply almost any make/model printer - please ask for details.

MONITORS - colour & mono - please call for details of our very latest offer
QL / Spectrum spares & repairs - please call for details

EEC Ltd
18-21 MISBOURNE HOUSE
CHILTERN HILL
CHALFONT ST PETER
SL9 9UE
TEL: 0753 888866

CWO, Access, VISA et cetera
Cheques - allow 10 days
Courier - £9 Postage - £3
Product subject to availability, E&OE.
FAX: 0753-887149

TF Services

MINERVA

The ULTIMATE operating system upgrade

MKII MINERVA now with battery for 256 bytes ram, CRASHPROOF clock & interface (I²C) for robots etc. Can autoboot from battery ram.

Other features (common to MKI/MKII):
Autoboot on reset/ Multiple Basic/ Faster scheduler-graphics(10% of Lightning)-string handling etc/ WHEN ERROR/ Second screen/TRACE Foreign kybd drivers/ 'Warm' fast reset etc

MKI to MKI UPGRADES: up to 1.79 - £5. Rest FREE (send rom (PROTECT LEGS!) and disk or 2 mdvs)

Original MKI..... £40 MKII.....£65

ALL VERSIONS FULLY GOLD CARD COMPATIBLE

QL SPARES

Faulty QL boards (excl plug in ICs)	£9		
Keyboard membrane	£8	Circuit diags	£2
68008 cpu	£8	1377 PAL	£3
JM rom set	£12	Power supply	£12
8302 ULA	£10	8301 ULA	£10
8049	£8	MDV ULA	£12

Other components (sockets etc)... plse PHONE

BULLETIN BOARD

QBBS - UK's first QL scrolling board

QL Echomail around Europe, and FIDO netmail
Around 10mb of files including the WHOLE King James' bible!

071-706 2379 V21/23/22/22bis 24hrs

QuaLsoft Terminal

Viewdata/VT52/VT100 QL TERMINAL EMULATOR

Multitasking for electronic mail, PRESTEL etc, CET download. Compatible with Tandata Qconnect QL modem/modaptor. Use Qconnect RS232C to drive unbuffered modems. Comprehensive A5 manual.

Software (3.5" or mdv) & manual £30

QL REPAIRS

QLs tested with Thorn EMI test rig and ROM software
(MDV hardware extra if required)

£27 - 6 month guarantee

COMPUTER CLEANER

HELP STOP POWER SUPPLY INDUCED LOCKUPS AND PROTECT COMPUTERS

Much improved mains filters. 40-80 db cut. Contain inductance, spike filters (three in 3 & 4 way) and certified 240vac capacitors.

2-way (5a adaptor) £14
3-way (5a adaptor) £18
4-way (13a trailing socket) £24

FILE TRANSFER

Programs to transfer text AND programs error free (XMODEM) between computers

Available for IBM PC and compatibles, ATARI ST and Sinclair QL. Connects to Psion organiser via Psion comms link

QuaLsoft program (per m/c) £7.50
Serial lead (name 2 computers) ... £10
Complete package for 2 computers. £25

All prices include VAT and postage in the UK

MAIL ORDER ONLY

12 Bouverie Place, London N2 1RB (tel: 071-724 9053)
SCROLLING BBS FOR ORDERING/QL HELP - 071-706 2379 (All speeds to V22bis)
Fax: 071-706 2379 Prestel: 017249853

Jochen Merz Software

Inst. str. Winkel 14 · W-4100 Duisburg 11 · Germany
☎ & Fax: 0203 501274

Jochen Merz Software is the largest supplier of high quality software that runs under QJUMP's Pointer Environment (PE). This hardware-independent environment is the very advanced and easy to use windowing system for QDOS, that is the path to the future for QL users. We support the PE by supplying software development tools, documentation and the QL Emulator

The QL-Emulator is an advanced replacement for the QL. It combines the power of ATARI's ST computers (which are very good value) with the extraordinary performance of the QL's operating system and the PE. More than 500 Emulators have been sold. 150 are used professionally.

For more details on our complete range of products write or send a fax!

QD III - The first (and only) Editor using the PE. Dynamic memory allocation, no limit on nrs of lines, comfortable block-handling and many, many features. **£38**

QMenu - the Menu Extension QMenu is an easy to use interface with predefined menus (e.g. file-select, simple-choice boxes, select from lists). These menus can be used from SuperBASIC, machine code and other languages. **£11**

FiFi - the FileFinder Running under the PE, FiFi is easy to use and extremely useful. FiFi scans devices or directory trees and searches files or filenames for strings. Combinations are possible, e.g. String1 AND string2 BUT NOT string3 etc. **£16**

DISA - Intelligent Disassembler which allows you to disassemble interactively any kind of machine-code. Completely pointer-controlled = very easy to use. Generates ASCII-output which may be assembled using the GST (Macro) Assembler. DISA also tries to insert label names (used in the Technical Guide and the Reference Manual), and also knows QDOS standards (e.g. Proc/Fn tables, ROM-headers, CONFIG-format etc.) **£29**

QDOS Reference Manual - This book is a must for all m-code programmers. It explains how to use QDOS, all traps and vectors, the Thing System, the HOTKEY System II and much more. It points out which features work on a QL, an Emulator and how to write compatible for future operating systems. DIN A5, 170 pages. £30

QPTR - The Pointer Environment Toolkit Revised manual which describes how to use the Pointer Interface and the Window Manager from SuperBASIC and machine code. QPTR comes with examples on disc, keys, macros and extensions for SuperBASIC. **£30**

FLP/RAM Level 2 - Replacement EPROM for SuperQBoards (V1.17 onwards, with & without mouse) and all TrumpCards. New Manual! Real sub-directories! QL-Emulator compatible! About twice as fast, with improved slave-block-handling! **£15**

EASYPTR II - Create your own PE menus & sprites, use them in your own SuperBASIC or machine-code programs. Creating Menus was not easy, until now! Put all the items you need on screen, that's it! Supports all the PE facilities, even split sub-windows and uses the Menu Extension! Now even complete application-sub-windows may be created! Many examples and better manual! NOW AVAILABLE! **£49**

QDesign - Graphic - and Design-program, also running completely under the PE! Supports the Extended Mode4. Do you think anything else is to say about a very good graphics program?! For 9, 18 and 24 pin printers **£38** or for Deskjet and Laserjet **£45**

Font - Vector - Editor for QDesign £19

QIMIs (Mouse-Interface for QL)
- with real-time-clock-battery-backup **£41** - without RTC (for GoldCard etc.) **£37.50**
Mouse for QIMI & ATARI ST £24
Mousepad, black, with QL's character set table £3.50
QL-Emulator Extended 4 for ATARI ST £166

Please add £4.00 for postage and package (Europe) or £4 or one item and £2.50 for every further item (Overseas).

All prices excl. V.A.T. E&OE.

Systematic Machine Code Programming

Part three of Alan Bridewell's modular machine code tutorial.

In this series, we are building up a library of small chunks of assembler language code. Each of these will do a recognisable job within a program, and be fully annotated, so that it is clear exactly how to join them to other chunks of code to make a program, and, in particular, how to make any necessary changes to make the chunks fit together.

In the first part, we looked at some simple chunks of code to enable us to get started and write a usable program. In the second part, we put together a simple, but useful, multitasking program.

Many programs that actually work well are thoroughly irritating to the user, and most of that irritation is caused by a poor screen display. Being able to control, and modify, the screen output, while the program is running, is the one major requirement for a professional, user-friendly screen display. This means being able to pan, scroll, or clear the screen (or part of it), change ink and paper colours, change window size, change border size and colour, change character size, precisely position the cursor, and draw lines and blocks on the screen.

Two groups

We can conveniently divide these up into two groups; the easy, and the not so easy. And what is it that makes some easy and others not? Answer: 'floating point arithmetic'. For most people programming in SuperBasic, a number is a number is a number. Any variable can be assigned any number (within the limits of the computer). The slightly more sophisticated will distinguish between 'Integer variables' (those ending in a '%' sign) and the others (usually referred to as 'floating point' or 'real' variables). Integer variables can only have whole number values over a limited range. They are supposed to compensate for this by being processed much faster by the QL. (In actual fact they're not. This is because the poor old QL spends so much time interpreting what the instructions actually mean, that you lose all the advantage of the faster processing. When you compile your SuperBasic, though, they are much faster.)

As far as the microprocessor is concerned, the difference is this. The floating point number is stored in a peculiarly coded form that needs Qdos routines to unravel. However, integer numbers are stored in a form that the microprocessor can load straight into its registers, and manipulate directly. Hence the speed and ease of use. In this article we shall restrict ourselves to the easy chunks of code requiring only integers.

These are all what we call TRAP #3 calls to Qdos. This means that the routines in

Listing 1

```
;*****************************************************************
;                        'WINDOWENQ'
;*****************************************************************
; THIS ROUTINE WILL FIND THE SIZE OF THE WINDOW AND POSITION OF THE CURSOR
; THE INFORMATION IS STORED IN 4 WORDS STARTING AT THE BASE OF ENQBLOCK.
; THE INFORMATION CAN BE RETURNED IN TWO FORMS, DEPENDING ON VALUE IN D0.
; D0 = $A           RETURN DATA IN PIXELS
; D0 = $B           RETURN DATA IN CHARACTERS. THE DATA IN THIS FORM
;          WILL VARY ACCORDING TO THE CSIZE AT THE CURSOR POSITION.
;
.WENQ               MOVE.L    (A7),A0    ; OR 4(A7),A0 OR 8(A7),A0, ETC.
                                           ;CONSOLE CHANNEL ID IN A0
                    LEA.L     ENQBLOCK,A1 ; BASE OF ENQBLOCK IN A1
                    MOVEQ     ##A,D0     ; #SD_PXENQ IN D0 (PIXELS)
                                         ; OR ##B,D0 #CHENQ IN D0 (CHARS)
                    MOVE.W    ##FFFF,D3  ; INFINITE TIMEOUT
                    TRAP      #3
                    BRA.S     NEXTBIT    ; SKIP BLOCK
;
; ** NOTE ** THE DATA IN A0 AND D3 IS PRESERVED BY THIS ROUTINE, SO IF THE
; NEXT CHUNK OF CODE IS ANOTHER TRAP #3 CALL FOR THE SAME CHANNEL, THEY
; DO NOT HAVE TO BE ENTERED AGAIN.
;
; BLOCK FOR THE RETURN OF WINDOW INFORMATION
;
.ENQBLOCK           DC.W      0          ; FOR X DIMENSION OF WINDOW
                    DC.W      0          ; FOR Y DIMENSION OF WINDOW
                    DC.W      0          ; FOR X POSITION OF CURSOR
                    DC.W      0          ; FOR Y POSITION OF CURSOR
;
;*****************************************************************
```

Listing 2

```
; *****************************************************************
;                          'BORDER'
; *****************************************************************
; THIS ROUTINE WILL REDEFINE THE BODER SIZE AND COLOUR IN THE WINDOW.
; ADJUST THE BYTE IN D1 FOR COLOUR, AND WORD IN D2 FOR BORDER WIDTH.
; ** NOTE ** DEFINING THE COLOUR AS ##80 WILL MAKE THE BORDER TRANSPARENT
; SO THAT ANY TEXT OR GRAPHICS FALLING WITHIN THE NEW BORDERS WILL NOT BE
; OVER-WRITTEN BY ANYTHING SUBSEQUENTLY WRITTEN IN THE WINDOW.
;
.BORD               MOVE.L    (A7),A0    ; OR 4(A7),A0 OR 8(A7),A0 ETC.
                                           ; CONSOLE CHANNEL ID IN A0
                    MOVEQ     ##C,D0     ; #SD_BORDER IN D0
                    MOVEQ     ##2,D1     ; BORDER RED
                    MOVE.W    ##10,D2    ; BORDER 16 PIXELS WIDE
                    MOVE.W    ##FFFF,D3  ; INFINITE TIMEOUT
                    TRAP      #3
;
; ** NOTE ** THE DATA IN A0 AND D3 IS PRESERVED BY THIS ROUTINE, SO IF THE
; NEXT CHUNK OF CODE IS ANOTHER TRAP #3 CALL FOR THE SAME CHANNEL, THEY
; DO NOT HAVE TO BE ENTERED AGAIN.
;
; *****************************************************************
```

Listing 3

```
; *****************************************************************
;                          'WDEFINE'
; *****************************************************************
; THIS WILL COMPLETELY REDEFINE THE SIZE, POSITION AND BORDER OF A WINDOW
; THE BORDER COLOUR IS A BYTE IN D1 AND BORDER WIDTH IS A WORD IN D2
; THE WINDOW SIZE AND POSITION IS IN A BLOCK STARTING AT WDEFINEBLOCK
;
.WDEF               MOVE.L    (A7),A0    ; OR 4(A7),A0 OR 8(A7),A0 ETC
                                           ; CONSOLE CHANNEL ID IN A0
                    LEA       WDEFINE,A1 ; WDEFINE BASE IN A1
                    MOVEQ     ##D,D0     ; #SD_WDEFINE IN D0
                    MOVEQ     ##2,D1     ; RED BORDER
                    MOVE.W    ##10,D2    ; BORDER 16 PIXELS WIDE
                    MOVE.W    ##FFFF,D3  ; INFINITE TIMEOUT
                    TRAP      #3
                    BRA.S     NEXTBIT    ; SKIP BLOCK
;
; ** NOTE ** THE DATA IN A0 AND D3 IS PRESERVED BY THIS ROUTINE, SO IF THE
; NEXT CHUNK OF CODE IS ANOTHER TRAP #3 CALL FOR THE SAME CHANNEL, THEY
; DO NOT HAVE TO BE ENTERED AGAIN.
;
; BLOCK CONTAINING WINDOW SIZE AND POSITION INFORMATION
;
.WDEFINE            DC.W      256        ; 256 PIXELS WIDE
                    DC.W      64         ; 64 PIXELS HIGH
                    DC.W      0          ; X ORIGIN
                    DC.W      0          ; Y ORIGIN
;
; *****************************************************************
```

Listing 4

```
; *****************************************************************
;                          'CURSOR'
; *****************************************************************
; WILL ENABLE OR SUPPRESS THE CURSOR IN THE WINDOW, DEPENDING ON THE
; VALUE IN D0
; D0 = $E           ENABLE CURSOR
; D0 = $F           SUPPRESS CURSOR
;
.CURSOR             MOVE.L    (A7),A0    ; OR 4(A7),A0 OR 8(A7),A0 ETC.
                                           ; CONSOLE CHANNEL ID IN A0
                    MOVEQ     ##E,D0     ; #SD_CURE IN D0 (ENABLE) OR
                                         ; ##F,D0 SD_CURS IN D0 (SUPPRESS)
                    MOVE.W    ##FFFF,D3  ; INFINITE TIMEOUT
                    TRAP      #3
;
; ** NOTE ** THE DATA IN A0 AND D3 IS PRESERVED BY THIS ROUTINE, SO IF THE
; NEXT CHUNK OF CODE IS ANOTHER TRAP #3 CALL FOR THE SAME CHANNEL, THEY
; DO NOT HAVE TO BE ENTERED AGAIN.
;
; *****************************************************************
```

Qdos are doing the real work. All we are doing is putting in the appropriate parameters, then telling Qdos to get on with it. They all require at least three parameters.

In register D0, it needs a byte to tell Qdos which routine to carry out.

In register D3, a word is needed for timeout (that is, how long the program will wait until the channel is ready to receive the command before it gives up trying to carry it out). With screen commands, I have yet to find a reason why timeout should not be infinite, that is, the program will wait until the command is successfully completed. The number for infinite timeout is -1 or #$FFFF in hex twos complement. (If you don't know what that is, don't worry about it for now. Your assembler will almost certainly deal with it for you.)

Channel ID

In register A0 we need the channel ID of the window we are dealing with. If this window was opened by the program, the ID will almost certainly be on the stack, quite possibly at the top. Frankly, manipulating a window from one program which has been opened by another (even #0, #1 and #2 from SuperBasic) means either rather crude programming, or you're attempting something really subtle! As a rule, every program should have its own windows, and leave all others alone. The only exception to this is with error messages from Qdos, which normally go to one of the SuperBasic windows. (After all, in the case of an error, your program may not have a window to print error messages!)

Some of these chunks of code require other numbers (for colours, sizes, distances, etc.) and some require the address of a parameter list, or some ram set aside for a reply, when Qdos has been asked for window information.

When I first embarked on the job of putting together this library of chunks, these TRAP #3 calls gave me much cause for thought and experiment. Basically, what we are dealing with is some 34 different calls to Qdos which do 34 slightly different things to the screen display. In many cases the difference in code is only one byte, so it seemed to be rather over the top to produce a separate library item for every call. On the other hand, to put them all together as one library item would mean adding a great deal of remarks to explain how to get it to do all the different jobs. It would be just like reading the manual again.

What I have done is group them together in 17 chunks which are similar enough to fit logically together (in my mind at least!).

'Windowenq' **(Listing one)** will find the size of the window and position of the cursor, and place them in a block of ram starting at the address in register A1. The information is returned as 4 words as follows:

At address in A1 X dimension of window
address in A1+2 Y dimension of window
address in A1+4 X position of cursor
address in A1+6 Y position of cursor

The information is returned either in pixels, or in characters, depending on the value in D0 – $A for pixels, and $B for characters. It is worth noting that if the information is returned in characters, it will depend on the Csize value for the character at the cursor position. A similar consideration occurs with other commands involving character co-ordinates.

'Border' **(Listing two)** will redefine the size and colour of the border. It requires a byte for the colour in register D1, and a word for the border width in register D2. The value required in register D0 is $C.

An interesting use of this is to make the border colour $80, which gives a transparent border, the text remains even when the rest of the text scrolls off the screen, or is removed by a CLS command. We shall make use of this in the next program we put together.

Window position

'Wdefine' **(Listing three)** will completely redefine a window (position, size, shape and border). This is very useful if you wish to position text or graphics just once in various parts of the screen without a succession of opening and closing of lots of channels. When the window is redefined, the contents of the old window position remain until they are over-written. Interesting results can be produced by making the new window partially overwrite the old.

The value required in register D0 is $D. As with 'Border', register D1 requires a byte for border colour and register D2 requires a word for border width. The window parameters must be stored in a parameter block the base address of which must be in register A1. The information is required as four words as follows:

Listing 5

```
; ***************************************************************
;                        'POSITIONCURSOR'
; ***************************************************************
; MOVES THE CURSOR TO AN ABSOLUTE POSITION IN THE WINDOW USING EITHER
; CHARACTER OR PIXEL COORDINATES, DEPENDING ON THE VALUE IN D0.
; D0 = $10          USE CHARACTER COORDINATES. REQUIRES A WORD FOR THE
;                   COLUMN NUMBER IN D1 AND A WORD FOR THE ROW NUMBER
;                   IN D2.
; D0 = $17          USE PIXEL COORDINATES. REQUIRES A WORD FOR THE
;                   X COORDINATE IN D1 AND A WORD FOR THE Y COORDINATE
;                   IN D2.
;
.CURS               MOVE.L    (A7),A0   ; OR 4(A7),A0 OR 8(A7),A0 ETC.
                                        ; CONSOLE CHANNEL ID IN A0
                    MOVEQ     #$10,D0   ; #SD_POS IN D0 (CHARS), OR
                                        ; #$17,D0 #SD_PIXP IN D0 (PIXELS)
                    MOVE.W    #$10,D1   ; COLUMN 16
                    MOVE.W    #$20,D2   ; ROW 32
                    MOVE.W    #$FFFF,D3 ; INFINITE TIMEOUT
                    TRAP      #3
;
; ** NOTE ** THE DATA IN A0 AND D3 IS PRESERVED BY THIS ROUTINE, SO IF THE
; NEXT CHUNK OF CODE IS ANOTHER TRAP #3 CALL FOR THE SAME CHANNEL, THEY
; DO NOT HAVE TO BE ENTERED AGAIN.
;
; ***************************************************************
```

Listing 6

```
; ***************************************************************
;                        'MOVECURSOR'
; ***************************************************************
; THIS MOVES THE CURSOR TO A NEW POSITION RELATIVE TO ITS OLD POSITION.
; THE MOVEMENT IS IN CHARACTER POSITIONS, AND DEPENDS ON THE VALUE IN D0.
; D0 = $12          PERFORMS A CR, LF, PUTTING CURSOR AT START OF NEXT LINE
; D0 = $13          MOVES CURSOR TO THE PREVIOUS COLUMN (LEFT)
; D0 = $14          MOVES CURSOR TO THE NEXT COLUMN (RIGHT)
; D0 = $15          MOVES CURSOR TO THE PREVIOUS ROW (UP)
; D0 = $16          MOVES CURSOR TO THE NEXT ROW (DOWN)
;
.MCURS              MOVE.L    (A7),A0   ; OR 4(A7),A0 OR 8(A7),A0 ETC.
                                        ; CONSOLE CHANNEL ID IN A0
                    MOVEQ     #$12,D0   ; #SD_NL IN D0 (CR,LF), OR
                                        ; #$13,D0 #SD_PCOL IN D0 (LEFT)
                                        ; #$14,D0 #SD_NCOL IN D0 (RIGHT)
                                        ; #$15,D0 #SD_PROW IN D0 (UP)
                                        ; #$16,D0 #SD_NROW IN D0 (DOWN)
                    MOVE.W    #$FFFF,D3 ; INFINITE TIMEOUT
                    TRAP      #3
;
; ** NOTE ** THE DATA IN A0 AND D3 IS PRESERVED BY THIS ROUTINE, SO IF THE
; NEXT CHUNK OF CODE IS ANOTHER TRAP #3 CALL FOR THE SAME CHANNEL, THEY
; DO NOT HAVE TO BE ENTERED AGAIN.
;
; ***************************************************************
```

Listing 7

```
; ***************************************************************
;                        'TAB'
; ***************************************************************
; MOVES THE CURSOR TO THE COLUMN NUMBER GIVEN BY THE WORD IN D1 IN THE
; CURRENT ROW.
;
.TAB                MOVE.L    (A7),A0   ; OR 4(A7),A0 OR 8(A7),A0 ETC.
                                        ; CONSOLE CHANNEL ID IN A0
                    MOVEQ     #$11,D0   ; #SD_TAB IN D0
                    MOVE.W    #$40,D1   ; TAB TO COLUMN 64
                    MOVE.W    #$FFFF,D3 ; INFINITE TIMEOUT
                    TRAP      #3
;
; ** NOTE ** THE DATA IN A0 AND D3 IS PRESERVED BY THIS ROUTINE, SO IF THE
; NEXT CHUNK OF CODE IS ANOTHER TRAP #3 CALL FOR THE SAME CHANNEL, THEY
; DO NOT HAVE TO BE ENTERED AGAIN.
;
; ***************************************************************
```

Listing 8

```
; ***************************************************************
;                        'CLS'
; ***************************************************************
; CLEARS ALL OR PART OF THE WINDOW.
; WHICH PART IS CLEARED DEPENDS ON THE VALUE IN D0.
; D0 = $20          CLEAR WHOLE WINDOW
; D0 = $21          CLEAR WINDOW ABOVE CURSOR LINE
; D0 = $22          CLEAR WINDOW BELOW CURSOR LINE
; D0 = $23          CLEAR THE CURSOR LINE
; D0 = $24          CLEAR CURSOR LINE TO THE RIGHT OF THE CURSOR,
;                   INCLUDING THE CHARACTER UNDER THE CURSOR
;
.CLS                MOVE.L    (A7),A0   ; OR 4(A7),A0 OR 8(A7),A0 ETC.
                                        ; CONSOLE CHANNEL ID IN A0
                    MOVEQ     #$20,D0   ; #SD_CLEAR IN D0  (ALL) OR
                                        ; #$21,D0 #SD_CLRTP IN D0 (TOP)
                                        ; #$22,D0 #SD_CLRBT IN D0 (BOTTOM)
                                        ; #$23,D0 #SD_CLRLN IN D0 (LINE)
                                        ; #$24,D0 #SD_CLRRT IN D0 (RIGHT)
                    MOVE.W    #$FFFF,D3 ; INFINITE TIMEOUT
                    TRAP      #3
;
; ** NOTE ** THE DATA IN A0 AND D3 IS PRESERVED BY THIS ROUTINE, SO IF THE
; NEXT CHUNK OF CODE IS ANOTHER TRAP #3 CALL FOR THE SAME CHANNEL, THEY
; DO NOT HAVE TO BE ENTERED AGAIN.
;
; ***************************************************************
```

At
address in A1 window width
address in A1+2 window height
address in A1+4 X origin
address in A1+6 Y origin

'Cursor' **(Listing four)** will either enable, or suppress, the cursor in the window, depending on the value in D0 ($E to enable, and $F to suppress).

'Positioncursor' **(Listing five)** will move the cursor to an absolute position in the window, using either pixel or character co-ordinates, depending on the value in D0 ($10 for characters and $17 for pixels). The word in D1 will give the new column number (or pixel X co-ordinate) and the word in D2 will give the new row number (or pixel Y co-ordinate).

Where 'Positioncursor' **(Listing six)** moves the cursor to an absolute position, 'Movecursor' makes a movement relative to the current position. The movement depends only on the value in D0 as follows:

$12 Put cursor at the start of the next line (NL,CR).
$13 Put cursor on the previous column (left).
$14 Put cursor on the next column (right).
$15 Put cursor on the previous row (up).
$16 Put cursor on the next row (down).

If the current position of the cursor makes the move impossible, the command is ignored, and the cursor stays where it was.

Overwriting

'Tab' **Listing seven** will move the cursor to the column number in register D1 in the current row. It requires #$11 in D0.

'Cls' **(Listing eight)** overwrites all, or part of, a window inside the current borders with the current paper colour. What gets overwritten depends on the value in D0 as follows:

$20 Clears the whole window.
$21 Clears the window above the cursor line.
$22 Clears the window below the cursor line.
$23 Clears the cursor line.
$24 Clears the right part of the cursor line up to, and including, the character under the cursor.

'Scroll' **(Listing nine)** scrolls all or part of the window by a distance given by the word in D1 (positive = down and negative = up). How much is scrolled depends on the value in D0 as follows:

$18 Scroll whole window.
$19 Scroll window above the cursor line.
$1A Scroll window below the cursor line.

'Pan' **(Listing ten)** pans a window sideways by a distance given by the word in D1 (positive = right and negative = left). How much is panned depends on the value in D0 as follows:

$1B Pan whole window.
$1E Pan the cursor line.
$1F Pan the right part of the cursor line up to, and including, the character under the cursor.

'Colours' **(Listing eleven)** will redefine the colour of the paper, strip or ink according to the byte in D1. Which it redefines

Listing 9

```
; ***************************************************************
;                          'SCROLL'
; ***************************************************************
; SCROLLS ALL OR PART WINDOW BY THE NUMBER OF ROWS GIVEN BY THE WORD IN
; D1. A POSITIVE NUMBER MOVES TEXT DOWNWARDS. A TEXT WINDOW WOULD NORMALLY
; BE SCROLLED UPWARDS, REQUIRING A NEGATIVE NUMBER.
; WHICH PART OF THE WINDOW IS SCROLLED DEPENDS ON THE VALUE IN D0.
; D0 = $18          SCROLL WHOLE WINDOW
; D0 = $19          SCROLL WINDOW ABOVE THE CURSOR LINE
; D0 = $1A          SCROLL WINDOW BELOW THE CURSOR LINE
;
.SCROLL             MOVE.L    (A7),A0   ; OR 4(A7),A0 OR 8(A7),A0 ETC.
                                        ; CONSOLE CHANNEL ID IN A0
                    MOVEQ     #$18,D0   ; #SD_SCROL IN D0 (WHOLE), OR
                                        ; #$19,D0 #SD_SCRTP IN D0 (TOP)
                                        ; #$1A,D0 #SD_SCRBT IN D0 (BOTTOM)
                    MOVE.W    #$FFFD,D2 ; SCROLL UP 3 ROWS
                    MOVE.W    #$FFFF,D3 ; INFINITE TIMEOUT
                    TRAP      #3
;
; ** NOTE ** THE DATA IN A0 AND D3 IS PRESERVED BY THIS ROUTINE, SO IF THE
; NEXT CHUNK OF CODE IS ANOTHER TRAP #3 CALL FOR THE SAME CHANNEL, THEY
; DO NOT HAVE TO BE ENTERED AGAIN.
;
; ***************************************************************
```

Listing 10

```
; ***************************************************************
;                           'PAN'
; ***************************************************************
; PANS ALL OR PART OF THE WINDOW BY THE NUMBER OF PIXELS GIVEN BY THE WORD
; IN D1. POSITIVE NUMBERS PAN RIGHT AND NEGATIVE NUMBERS PAN LEFT.
; WHICH PART IS PANNED DEPENDS ON THE VALUE IN D0.
; D0 = $1B          PAN WHOLE WINDOW
; D0 = $1E          PAN WHOLE CURSOR LINE
; D0 = $1F          PAN RIGHT END OF CURSOR LINE, INCLUDING CHARACTER
;                   UNDER THE CURSOR
;
;
.PAN                MOVE.L    (A7),A0   ; OR 4(A7),A0 OR 8(A7),A0 ETC
                                        ; CONSOLE WINDOW ID IN A0
                    MOVEQ     #$1B,D0   ; #SD_PAN IN D0 (ALL), OR
                                        ; #$1E,D0 #SD_PANLN (LINE)
                                        ; #$1F,D0 #SD_PANRT (RIGHT END)
                    MOVE.W    #$FFFD,D1 ; PAN LEFT BY 3 PIXELS
                    MOVE.W    #$FFFF,D3 ; INFINITE TIMEOUT
                    TRAP      #3
;
; ** NOTE ** THE DATA IN A0 AND D3 IS PRESERVED BY THIS ROUTINE, SO IF THE
; NEXT CHUNK OF CODE IS ANOTHER TRAP #3 CALL FOR THE SAME CHANNEL, THEY
; DO NOT HAVE TO BE ENTERED AGAIN.
;
; ***************************************************************
```

Listing 11

```
; ***************************************************************
;                          'COLOURS'
; ***************************************************************
; REDEFINES THE COLOURS THE WINDOW USES FOR:-
; PAPER (BACKGROUND CAUSED BY CLEAR, PAN OR SCROLL),
; STRIP (BACKGROUND BEHIND A PRINTED CHARACTER),
; INK (COLOUR CHARACTERS ARE PRINTED).
; WHICH IS REDEFINED DEPENDS ON THE VALUE IN D0.
; D0 = $27          REDEFINES PAPER COLOUR
; D0 = $28          REDEFINES STRIP COLOUR
; D0 = $29          REDEFINES INK COLOUR
; THE COLOUR IS GIVEN BY THE BYTE IN D1.
;
.COLOURS            MOVE.L    (A7),A0   ; OR 4(A7),A0 OR 8(A7),A0 ETC
                                        ; CONSOLE WINDOW ID IN A0
                    MOVEQ     #$27,D0   ; #SD_SETPA IN D0 (PAPER) OR
                                        ; #$28,D0 #SD_SETST IN D0 (STRIP)
                                        ; #$29,D0 #SD_SETIN IN D0 (INK)
                    MOVE.B    #7,D1     ; WHITE PAPER
                    MOVE.W    #$FFFF,D3 ; INFINITE TIMEOUT
                    TRAP      #3
;
; ** NOTE ** THE DATA IN A0 AND D3 IS PRESERVED BY THIS ROUTINE, SO IF THE
; NEXT CHUNK OF CODE IS ANOTHER TRAP #3 CALL FOR THE SAME CHANNEL, THEY
; DO NOT HAVE TO BE ENTERED AGAIN.
;
; ***************************************************************
```

Listing 12

```
; ***************************************************************
;                         'RECOLOUR'
; ***************************************************************
; THIS WILL CHANGE ALL THE COLOURS IN A WINDOW TO SOME OTHER COLOUR. IT
; REQUIRES AN 8 BYTE BLOCK WHICH CONTAINS THE BYTES FOR THE NEW COLOURS
; FOR COLOURS 0 TO 7 IN ORDER. IN 4 COLOUR MODE, ONLY BYTES 0, 2, 4 AND 6
; NEED BE SPECIFIED. THIS IS A RATHER SLOW ROUTINE. A MUCH FASTER
; ALTERNATIVE IS TO USE 'BLOCKFILL' OVER THE WHOLE WINDOW WHILE IN XOR
; MODE, BUT ONE IS LIMITED TO CHANGING COLOURS TO THEIR XOR VALUE.
.RCOL               MOVE.L    (A7),A0   ; OR 4(A7),A0 OR 8(A7),A0 ETC
                                        ; CONSOLE WINDOW ID IN A0
                    LEA       NCOLS,A1  ; NEWCOLOURS BLOCK BASE IN A1
                    MOVEQ     #$26,D0   ; #SD_RECOL IN D0
                    MOVE.W    #$FFFF,D3 ; INFINITE TIMEOUT
                    TRAP      #3
                    BRA.S     NEXTBIT   ; SKIP BLOCK
;
; ** NOTE ** THE DATA IN A0 AND D3 IS PRESERVED BY THIS ROUTINE, SO IF THE
; NEXT CHUNK OF CODE IS ANOTHER TRAP #3 CALL FOR THE SAME CHANNEL, THEY
; DO NOT HAVE TO BE ENTERED AGAIN.
;
; REDEFINITION BLOCK. ADJUST TO SUIT NEEDS
.NCOLS              DC.B      3         ; BLACK -> MAGENTA
                    DC.B      4         ; BLUE -> GREEN
                    DC.B      5         ; RED -> CYAN
                    DC.B      6         ; MAGENTA -> YELLOW
                    DC.B      7         ; GREEN -> WHITE
                    DC.B      0         ; CYAN -> BLACK
                    DC.B      1         ; YELLOW -> BLUE
                    DC.B      2         ; WHITE -> RED
;
; ***************************************************************
```

Listing 13

```
; ***************************************************************
;                          'FLASH'
; ***************************************************************
; SETS THE FLASH IN 8 COLOUR MODE. IF THE BYTE IN D1 IS NON-ZERO, THEN
; FLASH IS SET. A ZERO BYTE IN D1 MEANS FLASH IS NOT SET.
; THIS COMMAND IS IGNORED IN 4 COLOUR MODE.
;
.FLASH              MOVE.L    (A7),A0   ; OR 4(A7),A0 OR 8(A7),A0 ETC
                                        ; CONSOLE WINDOW ID IN A0
                    MOVEQ     #$2A,D0   ; #SD_SETFL IN D0
                    MOVE.B    #1,D1     ; FLASH SET
                    MOVE.W    #$FFFF,D3 ; INFINITE TIMEOUT
                    TRAP      #3
;
; ** NOTE ** THE DATA IN A0 AND D3 IS PRESERVED BY THIS ROUTINE, SO IF THE
; NEXT CHUNK OF CODE IS ANOTHER TRAP #3 CALL FOR THE SAME CHANNEL, THEY
; DO NOT HAVE TO BE ENTERED AGAIN.
;
; ***************************************************************
```

Listing 14

```
; ***************************************************************
;                         'UNDERLINE'
; ***************************************************************
; SETS UNDERLINE MODE FOR PRINTED CHARACTERS. IF THE BYTE IN D1 IS NON-
; ZERO, CHARACTERS WILL BE UNDERLINED. IF THE BYTE IS ZERO, THE CHARACTERS
; WILL NOT BE UNDERLINED
;
.UNDER              MOVE.L    (A7),A0   ; OR 4(A7),A0 OR 8(A7),A0 ETC
                                        ; CONSOLE WINDOW ID IN A0
                    MOVEQ     #$2B,D0   ; #SD_SETUL IN D0
                    MOVE.B    #1,D1     ; SET UNDERLINE MODE
                    MOVE.W    #$FFFF,D3 ; INFINITE TIMEOUT
                    TRAP      #3
;
; ** NOTE ** THE DATA IN A0 AND D3 IS PRESERVED BY THIS ROUTINE, SO IF THE
; NEXT CHUNK OF CODE IS ANOTHER TRAP #3 CALL FOR THE SAME CHANNEL, THEY
; DO NOT HAVE TO BE ENTERED AGAIN.
;
; ***************************************************************
```

depends on the value in D0 as follows:

$27 Redefine paper colour
$28 Redefine strip colour
$29 Redefine ink colour.

'Recolour' **(Listing twelve)** will recolour every pixel in the window to a new colour given by an 8-byte conversation list, the base of which has its address in A1. It requires $26 in D0.

This is a very flexible command, but is also slow. A much faster (but somewhat restricted) alternative is explained under 'Blockfill'.

Flashing

'Flash' **(Listing thirteen)** only works in 8 colour mode. If the byte in D1 is non-zero, then subsequently printed characters will be flashing, until the byte in D1 is made zero, when any further characters will not be flashing. Requires $2A in D0.

If the byte in D1 is non-zero, 'underline' **(Listing fourteen)** makes subsequently printed characters underlined, until the byte in D1 is made zero, when any further characters will not be underlined. It requires $2B in D0.

'Writemode' **(Listing fifteen)** redefines how the strip and ink colours are interpreted on the screen, depending on the word in D1 as follows:

-1 Ink colour is XORed into the background.
0 Normal printing in ink colour on strip colour.
1 Printing in ink colour on a transparent strip which does not overwrite the background.

It requires $2C in D0.

'Csize' **(Listing sixteen)** (surprise, surprise!) redefines the character size. It requires $21 in D0, and will set the width depending on the word in D1, and the height depending on the word in D2, as follows:

In D1

0 5 pixel width (single) in 6 pixel space.
1 5 pixel width (single) in 8 pixel space.
2 10 pixel width (double) in 12 pixel space.
3 10 pixel width (double) in 16 pixel space.

in D2

0 9 pixel height (single) in 10 pixel space.
1 18 pixel height (single) in 20 pixel space.

It should be noted that in 8-colour mode, only double-width characters are allowed, and the QL will interpret a 0 in D1 as a 2, and a 1 in D1 as a 3.

'Blockfill' **(Listing seventeen)** will fill a rectangular block inside the windows with a colour given by the byte in D1. It requires $2E in D0, and in A1 the base address of 8 bytes containing the block details as follows:

Base Width in pixels.
Base + 2 Height in pixels.
Base + 4 X Origin of block (relative to window).
Base + 6 Y Origin of block (relative to window).

There are two special applications of this command that make it particularly useful because of its speed.

A long, thin block will draw a horizontal or vertical line much more quickly than normal line plotting will.

If 'Writemode' is set to XOR mode, then all the colours currently in the block will be XORed with the colour given by the byte in D1. Although this is not a completely general recolour command, it gives quite a lot of choice, and is much faster than 'Recolour', particularly when handling large areas of the screen.

Now to try out some of this. Clearly, an application which used all of these chunks of code would be very unusual indeed! Perhaps you could try modifying the program we put together in part two, by making the display it produces more user-friendly, or prettier, or more bizarre if that's what turns you on!

Interactively

I expect, though, that if you are going to try out this material, you probably want to do it interactively, like Basic. Like most people brought up to do interactive Basic, I write a bit, test it, write a bit more, test it. . . etc. In Basic, this presents no problem, because if it doesn't do what you wanted, the worst that happens is an error message.

Machine code programs that don't work properly usually crash the machine, or at least leave you with lots of unwanted open channels clogging up the system so it needs a reset.

The trick of writing interactively is to write the start and finish first, that is 'JOBSTART' and 'ENDJOB'. A program consisting of just this won't do anything, but it won't crash the QL either. Next, in between those, we put 'CONSOLE' to open a window, and 'CLOSE' to close it. Now we have a 'do nothing' channel, but still it won't crash. Now, in between 'CONSOLE' and 'CLOSE' we can do what we like with the window, and provided we get all the BRA.S command right to make it a continuous program, we can play around until we get what we want.

Next time, we shall look at some of the things we can do with these screen control commands to produce interesting effects.

Listing 15

```
; ***************************************************************
;                        'WRITEMODE'
; ***************************************************************
; REDEFINES HOW THE CHARCTERS AND BACKGROUND ARE TO BE PRODUCED BY THE
; WORD IN D1.
; D1 = -1 INK IS XOR CURRENT BACKGROUND
; D1 =  0 WRITE IN INK COLOUR ON STRIP COLOUR BACKGROUND
; D1 =  1 WRITE IN INK COLOUR ON A TRANSPARENT BACKGROUND
;
.WMODE          MOVE.L    (A7),A0    ; OR 4(A7),A0 OR 8(A7),A0 ETC
                                     ; CONSOLE WINDOW ID IN A0
                MOVEQ     #$2C,D0    ; #SETMD IN D0
                MOVE.W    #$FFFF,D1  ; XOR INK INTO BACKGROUND
                MOVE.W    #$FFFF,D3  ; INFINITE TIMEOUT
                TRAP      #3

;
; ** NOTE ** THE DATA IN A0 AND D3 IS PRESERVED BY THIS ROUTINE, SO IF THE
; NEXT CHUNK OF CODE IS ANOTHER TRAP #3 CALL FOR THE SAME CHANNEL, THEY
; DO NOT HAVE TO BE ENTERED AGAIN.
;
; ***************************************************************
```

Listing 16

```
; ***************************************************************
;                        'CSIZE
; ***************************************************************
; REDEFINES THE CHARACTER SIZE AND SPACING.
; THE WORD IN D1 SETS THE CHARACTER WIDTH/SPACING AS FOLLOWS:-
; D1 = 0  SINGLE WIDTH (5 PIXELS) IN 6 PIXEL SPACE
; D1 = 1  SINGLE WIDTH (5 PIXELS) IN 8 PIXEL SPACE
; D1 = 2  DOUBLE WIDTH (10 PIXELS) IN 12 PIXEL SPACE
; D1 = 3  DOUBLE WIDTH (10 PIXELS) IN 16 PIXEL SPACE
; 8 COLOUR MODE CAN ONLY SUPPORT DOUBLE WIDTH CHARACTERS, SO D1 = 0 WILL
; BE TREATED AS D1 = 2, AND D1 = 1 WILL BE TREATED AS D1 = 3.
; THE WORD IN D2 SETS THE CHARACTER HEIGHT/SPACING AS FOLLOWS:-
; D2 = 0  SINGLE HEIGHT (9 PIXELS) IN 10 PIXEL SPACE
; D2 = 1  DOUBLE HEIGHT (18 PIXELS) IN 20 PIXEL SPACE
;
.CSIZE          MOVE.L    (A7),A0    ; OR 4(A7),A0 OR 8(A7),A0 ETC
                                     ; CONSOLE WINDOW ID IN A0
                MOVEQ     #$2D,D0    ; #SD_SETSZ IN D0
                MOVE.W    #$2,D1     ; WIDTH 10 IN 12 PIXEL SPACE
                MOVE.W    #$1,D2     ; HEIGHT 18 IN 20 PIXEL SPACE
                MOVE.W    #$FFFF,D3  ; INFINITE TIMEOUT
                TRAP      #3

;
; ** NOTE ** THE DATA IN A0 AND D3 IS PRESERVED BY THIS ROUTINE, SO IF THE
; NEXT CHUNK OF CODE IS ANOTHER TRAP #3 CALL FOR THE SAME CHANNEL, THEY
; DO NOT HAVE TO BE ENTERED AGAIN.
;
; ***************************************************************
```

Listing 17

```
; ***************************************************************
;                        'BLOCKFILL'
; ***************************************************************
; THIS FILLS A RECTANGULAR BLOCK WITHIN THE WINDOW.
; THE COLOUR IS GIVEN BY THE BYTE IN D1.
; THE DATA FOR THE BLOCK TO BE FILLED IS IN AN 8 BYTE DEFINITION BLOCK
; CALLED 'BLOCK' AS FOLLOWS:-
; BASE              BLOCK WIDTH IN PIXELS
; BASE + 2          BLOCK HEIGHT IN PIXELS
; BASE + 4          X ORIGIN OF BLOCK (RELATIVE TO WINDOW ORIGIN)
; BASE + 6          Y ORIGIN OF BLOCK (RELATIVE TO WINDOW ORIGIN)
; THE BASE ADDRESS OF BLOCK MUST BE IN A1.
;
.BFILL          MOVE.L    (A7),A0    ; OR 4(A7),A0 OR 8(A7),A0 ETC
                                     ; CONSOLE WINDOW ID IN A0
                LEA       BLOCK,A1   ; BASE ADDRESS OF BLOCK IN A1
                MOVEQ     #$2E,D0    ; #SD_FILL IN D0
                MOVE.B    #$2,D1     ; RED BLOCK
                MOVE.W    #$FFFF,D3  ; INFINITE TIMEOUT
                TRAP      #3
                BRA.S     NEXTBIT    ; SKIP BLOCK

;
; ** NOTE ** THE DATA IN A0 AND D3 IS PRESERVED BY THIS ROUTINE, SO IF THE
; NEXT CHUNK OF CODE IS ANOTHER TRAP #3 CALL FOR THE SAME CHANNEL, THEY
; DO NOT HAVE TO BE ENTERED AGAIN.
;
; ** NOTE ** APART FROM SIMPLY FILLING A BLOCK WITH A COLOUR, THIS CAN BE
; USED IN TWO OTHER WAYS.
; 1. LONG THIN BLOCKS WILL DRAW HORIZONTAL OR VERTICAL LINES.
; 2. WITH WRITEMODE SET TO 'XOR', IT WILL RECOLOUR A BLOCK OR WINDOW.
; THESE HAVE THE ADVANTAGE OF BEING FASTER THAN THE CONVENTIONAL WAYS.
;
.BLOCK          DC.W      256        ; 256 PIXELS WIDE
                DC.W      64         ; 64 PIXELS HIGH
                DC.W      0          ; AT THE WINDOW ORIGIN
                DC.W      0

;
; ***************************************************************
```

SPECIAL OFFER TO ALL OVERSEAS READERS

We are at present experiencing difficulties in exporting copies of Sinclair QL World, So that you do not miss any issues, we invite you to take advantage of the following offer:

Why not buy yourself a subscription to your favourite magazine for Christmas

Subscribe for one year and you will recieve 15 issues for the price of 12

that's 3 issues free!

Subscribe for two years and get 3 free issues and a discount of £4.00!

(See special discount rates below)

Fill in the coupon below and send it with your remittance to: MSM Subscriptions Department, Lazahold Ltd., P.O. Box 10, Roper Street, Pallion Industrial Estate, Sunderland SR4 6SN. (The first issue of a new subscription to be delivered will be one or two issues after the one you placed your order in).

Please start/renew my 12/24 month subscription to SQLW. I will receive 3 free issues before my subscription becomes due for renewal. I enclose my cheque my cheque/money order for £_____ made payable to MCPC Ltd or debit my Access/Visa card.

No._____ Expiry date_____

Name._____

Address._____

_____

_____ (Please enter postcode to ensure prompt delivery)

Signed _____ Date_____ SQLW 0192

Europe £32.90 12 mth
£59.80 24 mth

Rest of £40.90 12 mth
World £87.80 24 mth

Overseas rates include Airmail Service

BOOK PAGE

Title: *Professional and Business Uses of the QL*
Author: Colin Lewis
Publisher: Collins Profesional, 1985
Price: £7.95 in 1985
Available: Fairs and second-hand
QL Specific: Yes

At its launch the QL was seen by many small businessmen and self-employed professionals as the very thing they needed – a work-orientated, affordable, small computer that could bring more efficient handling of paperwork.

At the same time these QL buyers were far too busy to spend weeks learning to program – or even to wade through the heavy instruction manual supplied with the machine and it was for these users that Colin Lewis brought out his book *Professional and Business Uses of the QL* in 1985. The book cost £7.95 at that time and was published by Collins.

The book didn't claim to say anything really new about the QL. It simply concentrated on the bundled software package supplied with the machine and showed how it could be put to immediate use in a business environment.

At that time I was a freelance journalist looking to replace an ageing typewriter with a word processor and to develop a computer-based filing system to keep track of my work placement and this book very soon became my QL bible.

The book began with the setting up of the hardware and its linkage to an Epson-type printer and, as it was dealing with the QL as supplied, all the practical applications listed in the book were set up to be run using microdrives. After the setting up procedures had been explained the book outlined the very minimum knowledge that was needed to use Qdos and then launched into the practical applications of *Quill*.

All the author needed to do here, of course, was to show how a wordprocessor could replace the office typewriter and he finished up his demonstration with a sample price quotation using right and decimal tab stops. The next chapter, dealing with *Abacus*, gave several practical examples of the spreadsheet's use – including building an Investment Portfolio, a Break-even Analysis spreadsheet and also one dealing with cash flow analysis and the production of a document that could be presented by the director of a small company to his bank manager when needing loan facilities.

Easel, which the author regards as quite a useful business program, is covered very thoroughly in the book but, for me, his explanation of *Archive* and its practical business applications was the most useful thing of all. It is quite amazing how many QL owners never use Archive because they are convinced it is over complicated. Although I am strictly a non-technical person I found myself building up some highly sophisticated databases by following the examples in this book. For some it could be worth getting the book for this application alone.

For any businessman already using the four packaged programs on a regular basis, however, this book would probably be of little value at the present time. If a similar book could be written today outlining the business possibilities of a QL which has an add on disk-drive and extended memory and for which several business and accounts programs have been produced — now, that really would be useful.

David Drysdale

Elementary

Title: *Exploring the Sinclair QL – An Introduction To SuperBasic*
Author: Andrew Nelson
Publisher: Interface Publications, 1984
Price: £4.95 in 1984
Available: Fairs, or second-hand
QL Specific: Yes

This book is just what it calls itself: an introduction to SuperBasic and an elementary one at that. I bought it at the time when the first functioning (ie AH) QLs were being unveiled to the public. Not surprisingly then, the book carries a computer health-warning being 'based on use of pre-release QL computers', and goes on to caution the reader to check the *User Guide* if problems arise. For example, Nelson describes how to get the Ascii code of the character "A" by the statement: PRINT CODE "A" instead of PRINT CODE ("A"), although he does get it right in an Appendix.

Following a brief tour of the QL hardware, the greater part of the book describes the range of SuperBasic commands available and their use in programming. Nelson is a firm believer in the merits of the structured approach. In general, his descriptions are clear and concise without getting bogged down in unnecessary detail.

His presentation of the main elements is logical, starting with simple commands, then introducing control statements, arrays and strings, procedures and functions, and finally graphics. Towards the end, there is a very sketchy account of Qdos and channels.

In some ways, the book reads like an account of the SuperBasic we might have had with, for example, a fascinating description of array literals, edit-mode overwriting, and keywords such as INVERSE and TRACE which were never implemented by Sinclair Research. Also, the syntax of a number of keywords has changed since the earliest versions were released, a fact which rather lets the book down.

I found the discussion of how screen colour output is controlled muddled and confusing and also his treatment of output 'devices' and channels which, rather unfortunately, he refers to as pipes. The example programs given throughout the book are extremely elementary, none of them doing anything at all useful.

There are a few niggles I have about the book as well, eg, the blank, but numbered, pages between chapters giving the impression of sections being left out; the large number of space-wasting photographs of the QL (including one of the infamous 'kludge' or 'dongle'), the unnecessary discussion of monitor-lead pin-connections on pages 4 and 5, not to men-

tion the jaded innuendo in the section on graphics.

With its unfussy approach and clear style, the book has some merit as a simple introduction to programming in SuperBasic. However, there are far too many inaccuracies in it to recommend it any more highly than that.

Nigel Bates

From Uppsala to Apple

Title: *Electronics Handbook*
Author: Jorge de Sousa Pires
Publisher: Chartwell-Bratt
Price: £39.95
Available: Currently in print
QL Specific: No

This book of just over eight hundred pages covers a wide range of electronics and related subjects. It includes some relevant basic physics and mathematics, analogue circuits, digital circuits, computer hardware and computer software.

The author was associate professor of semiconductor physics at Uppsala University for 15 years, and now works for Apple Computer. No doubt this experience explains the tremendous range which he covers convincingly.

The book is aimed at students, and it is structured to enhance its function as a textbook. Subject areas are split into small sections, which are not contiguous. A section typically covers a limited subject area from basic to advanced level, and may include notes on what the teacher/lecturer should do to supplement the information in the book.

The author has included a section on 'Pedagogics' in which he explains among other things that the book is intended to promote understanding of the subjects rather than simply teach the relevant mathematics. It should achieve this aim. Certainly the sections which I studied in detail made good sense and were as straightforward as the subject matter allowed.

The book is well indexed for its use as a textbook, but not so well for use as a reference book.

In addition to areas normally covered by this kind of book, the author has included some short notes on related subjects such as astronomy and music. Peripheral subjects are also touched on in individual sections, so I learnt when reading about signature analysis for testing microprocessors, that the word comes from the Latin - *signatura* – the marking of sheep for identification. So you can count microprocessors instead of sheep.

This is not a specialised computer book. Most long term QL users probably already know a large proportion of what the computer section contains, and considerably more about software. The subject is covered sufficiently well, however, that almost everyone will discover something new to them. Among *QL World* readers, those interested in general electronics should look this book over. It will provide a worthwhile depth of information for people interested in how their computers function as well as how to program them.

On the computing front, there are many examples of Basic programs included in other sections. Bode plots and calculation of conic sections are two examples of this. There is also a section about how to write better and faster Basic (not, of course, specifically SuperBasic). Operating systems and assemblers are also covered in short sections, but these are only introductions. Most computer topics are at least touched upon, with a worthwhile section on modems as one example.

Spreadsheet calculations are also used as examples, and typical printouts show an Apple-based package. Most or all of the calculations would be possible on other spreadsheets, though, with only some of the graphics not being universally available.

My only specific criticism is that, in covering digital circuits, the author has used the unpopular IEC logic symbols, in which all gates are drawn as rectangles. Symbols written inside the rectangle define the gate type, so that the functioning of a logic circuit is not clear at a glance, as it usually is when using the ANSI (American) symbols.

Taking a final lucky-dip of subject headings from the book to give an idea of its scope, we come across decibels, scientists and historical facts, noise reduction, astronomy, Laplace transform, vectors, Boolean algebra, debouncing, Morse code, DTMF tones, world tv and Mesh analysis alongside the more obvious electronics, computing and comms headings.

The *Electronics Handbook* would also be a useful sourcebook for anyone studying for a qualification in electronic or electrical engineering, or for electronics specialists needing a reasonable depth of background knowledge outside their specialism.

The price will tend to rule it out for the idly curious, but the serious student or enthusiast will appreciate it.

Andrew Armstrong

Worlds from Scientific American

Title: *The Armchair Universe*
Author: A K Dewdney
Publisher: W H Freeman, Oxford
Price: £12.95
Available: Currently in print
QL Specific: No

As I learned how to program a computer, there often came times when I would consider exactly which problems I might choose to solve with my new found skills. I would derive as much pleasure from small programs that would perform simple jobs as I might from far more complex (and often long since abandoned) projects. Tasks such as attempting to find meaningful anagrams of words – or indeed even ones that could be pronounced at all provided both entertainment and improved my ability to write programs.

One useful source of ideas for such programs was, and indeed still is, *Scientific American*. Anyone who has found interest in Mandelbrots, artificial intelligence, the game of 'Life' and similar computer related activities is likely at one time to have read A K Dewdney's *Computer Recreations* columns in this magazine.

This book represents a collection of the subjects detailed by Dewdney in his articles. The first to be produced, it recounts in just over three hundred pages, some of the topics covered between 1984 and 1987.

The style is light and highly illustrative, with most themes being introduced through imaginative descriptions of the processes involved. It is not to be read through from beginning to end, though. The many, often wildly different ideas are best picked up individually when their descriptions have aroused your interest.

To this end, the book is divided into sections – different 'Worlds' – each with its own introduction. These include the worlds of Infinite Graphics, Artificial Intelligence and Mathemagadgets amongst others – which in turn take in respectively Mandelbrots, the well known Eliza program and devices made of spaghetti. Each world covers a number of subjects which are clearly described and illustrated, either in diagrams or the central colour pages. The descriptions take care not to become too complex, nor to use mathematics beyond basic algebra. In some cases, where concepts that might be new to the reader are introduced, it might help to have a pencil and paper to hand – however Dewdney takes these descriptions slowly and treats them with a disarming humour.

If, as is often the case, a computer program might be developed from the discussion, this too is described. The descriptions have no listings within them (bar one itinerant Apple program), but instead list each step that must be per-

formed in turn. The process of translating this into a program is relatively simple and in any case best left to the reader, so as to suit his particular computer set-up.

Other worlds covered by the book include 'Life In Automata', 'Puzzles and Wordplay', 'Stimulation through Simulation' and 'Core Wars' – the last being the fascinating area of battling computer programs. The sections are complemented by a full index, a bibliography and a list of sources of further information and software (the latter being largely based in America and intended for the IBM PC). Overall, the book is well thought out and written. It presents any number of both new and old topics very clearly and is to be recommended.

Andrew Toone

C: Portable from the general to the specific

Title: *C: A Dabhand Guide*
Author: Mark Burgess
Publisher: Dabs Press
Price: £14.95 paperback
Available: Currently in print, other books in series
QL Specific: No

This is one of a series of computer books set out in a similar format, and is primarily intended as a tutorial, with only a slight knowledge of computing in a language such as Basic assumed. The book starts with the basic concepts of high and low-level languages, the differences between compilers and interpreters, progressing in easy stages with plenty of example listings and diagrams as new keywords and concepts are introduced.

Some of the topics covered are Pre-processor, Libraries, Parameters, Local and Global variables, Pointers, Functions and Macros, Bit Operators, Data Structures and Recursion. Don't be put off if you have not heard of some of these terms, they are introduced gradually, and each chapter finishes with a series of questions to check your understanding of the chapter – the answers are in one of the appendicies at the back of the book.

The examples culminate towards the end of the book in a listing of useful toolkit routines covering console input and output, complex numbers, and linked list data structure. There are also two complete program listings, a statistical data handler utility (calculates the mean, standard deviation etc. from sets of data), and a variable cross referencer (produces a list of all the identifiers – variables etc. with line numbers within a source file.

The index is fairly extensive so when the language is mastered the book would still be useful for reference. Of necessity the text is not machine specific. (C is meant to be a very portable language) however towards the end of the book there are chapters detailing the peculiarities of C in relation to several machines Amiga, Atari ST, Archimedes, PC, and BBC Master, though unfortunately nothing specific to the QL.

There are program disks available to accompany this book in the same five formats as mentioned above. The publisher has informed me that the current price is £7.95 but further information can be obtained from Dabs Press on 061 773 8632 (this is a more recent number than that given in the book).

I have been working through this book using DP's *PC Conqueror* with the shareware program *Personal C Compiler* to compile the examples without any major problem. I have also tried the listing using *Digital C Special Edition* compiler with less success, as Digital C is only a subset (based on Small-C) of the complete C language.

C: A Dabhand Guide is a well presented and easy to follow tutorial text that can later be used for reference.

John Langford

Words for people who like getting to the heart of things

Title: *Inside the Sinclair QL – an introductory guide to the hardware*
Author: Jeff Naylor and Diane Rogers
Publisher: Sunshine Books (Scot Press) 1985
Price: £6.95 in 1985
Available: Fairs or second-hand
QL Specific: Yes

Most computer users are one of two types: the timid sort content to run someone else's software which does what they want without anything going seriously wrong, and the sort who are fully conversant with machine code, instruction sets, and microprocessor architectures and who see computers as things to be taken apart, debugged and reassembled like the parts of a car.

Nonetheless, I suspect there are numerous QL owners who long for the day when they might leap the gap between novice user and informed expert. If you're one of these, then this book is as good a place to start as you could find. Naylor begins with the most elementary principles of electronics and builds up to a fairly detailed description of the QL architecture and hardware, and the Motorola 68008 instruction set.

The book is divided into two sections, the first of which deals with the fundamental principles of electricity, digital circuits, logic gates, sound and image production, and magnetic recording of data. After a clear and concise account of number bases, the book illustrates the main components of a hypothetical microcomputer architecture: central processing unit, buses, memory devices, keyboard, and video circuits.

Then follows a detailed look at the inner workings of a typical cpu through its various buses, registers, program counter, stack pointer, and the way in which the cpu obeys instructions, fetches data, addresses memory and other such operations.

The second section describes the detailed make-up of the QL, in relation to the principles already outlined. A brief account of the 68008 processor itself leads into an extensive reference section on the 68008 instruction set. The author takes the MOVE instruction as an example to describe such things as syntax, opcode and the effect of the instruction of the various flags.

From here, the book explores machine code in a few simple routines which, however, only succeeded in locking up my (expanded) QL. Naylor includes a simple monitor program to examine and alter memory addresses, save and call code and carry out number-base conversions.

There are detailed descriptions of the physical and software memory maps, the video display, the 8049 processor, keyboard, CTL and serial sockets, sound, a very skimpy section on networking, and a description of the microdrive filing system.

Finally, in a chapter entitled 'Software Maketh the Machine', the author describes Qdos traps and vectors, and exhorts the reader to try out things for himself.

The text is well written, thanks also to the literary contribution of Diane Rogers which the blurb on the back cover acknowledges. There were a few occasions, however, when a picture (or two) certainly would have been worth a thousand (or $3E8!) words. Having said that, I will admit that I did come away with a better feel, at least, for how my QL ticks.

However, you'll need to read this book more than once if this is your first attempt at breaking into the inner circle of computer aficionados. In those other immortal words of Abraham Lincoln, 'People who like this sort of thing will find this the sort of thing they like.'

Nigel Bates

MICRO ADS
MICRO ADS
MICRO ADS
MICRO ADS
MICRO ADS
MICRO ADS

FOR SALE. QL, Gold Card, Green Monitor, Real Time Digitizer, Twin 3½" Disk drives. Software: Lightning, Editor, Turbo, Assembler, Digital C; Conqueror; Chess Touch Typist. All for £750. Tel: 021 449 8041.

PENMAN PLOTTER, TECHNIQL & TECHNIKIT CAD PACKAGE. Includes 3-colour plotter with turtle graphic capability, font designer, & printer driver. Boxed complete with manuals, pens & leads. Only £180. Phone Sam – 051-728 7196 (after 7pm).

QL ROMS AND SPARES

ROMS, MGE (Spanish, MGF (French), MGG (German), JS (English), £20 each. JM (English), £10

MDVS. Complete QL MDV Subunit £20, head and chassis £10, Microswitch £2, Motor £6. Rubber & Plastic Roller £2. MDV2 Stand-off spacer + srew £2. ULA £5. Tr 4,5,6,7 set £2. MDV pcb inc, ULA £11. Prices include UK postage. Send SAE for price list of all QL spares.

Joe Atkinson, 36 Ranelagh Road, Ealing, London W5 5RJ (Mail Order Only).

QL GAMES. £13, inclusive, menu selectable cribbage, solitaire, Pairs, Bingo. Send cheques payable to AP Etheridge, 30 The Heights, Foxgrove Road, Beckenham, Kent, BR3 2BY. (Send MDV if disk not suitable).

QL REAL TIME DIGITIZER

Digitize moving pictures in real time. Transforms QL into a video printer or 3-D image digitizer via a video camera/recorder

* Digitizes in .02 secs.
* 256x256 resolution in 8 levels
* Full, ¼ or 4x ¼ screen modes
* Screen update - 5 or 2.7 frame/sec
* 1:1 grey level 9-pin printer dump.
* Interfaces to QL ROM PORT
* Runs on unexpanded QL and multitasks.
* Cost: £99.00 incl. State mdv or flp.

£99

CL SYSTEMS, 403 CHAPTER ROAD, DOLLIS HILL LONDON NW2 5NG TEL: 081 459 1351

FOR SALE. Complete QL business system, Computer + 768k Trump Card mounted in Quest Executive case, twin 5.25" drives, colour monitor, autodial/receive modem, Epson D/wheel printer and sheet feeder. Software incs. Lightening S/E, Conqueror + DOS, Mailmerge, Flashback S/E, Discover, Taskmaster, Editor2, Spellbound, Q-Connect, PSION etc. Everything you need in a serious system for just £700 ono. Telephone(0705) 670550 or 652011

FOR SALE. QL (JM) 768K Trump Card, twin 3.5" NEC drives, monitor, software (Taskmaster, Assembler/ Debugger), books, microdrive cartridges, floppy discs. £400 or nearest offer. Tel: 0902 871030 9am-7pm

QL to PC transfer service. We will transfer your QL DBF/Doc files to PC format. Doc files £10.00 up to 50k, DBF files £15.00 up to 100k. Reduced rates for larger files. DBF files to your specification. PRG/Pro files. P&P and other charges £1.50. S & P Computing Services, 188 Copley Close, St Johns, Woking, Surrey GU21 1TW. Tel: 04867 80241

QL DISASSEMBLER £10; MS-DOS Reader, £5.00. Kevin Robinson, Utility Room, 5 Upton Way, Walshaw, Bury, Lancs, BL8 3BT.

CLASSIFIED

Please send coupon to: Advertising Department Sinclair/QL World, Panini House, 116/120 Goswell Road, London EC1. Pleae place this Classified Advertisement in the next issue of Sinclair/QL World.

NAME .....

ADDRESS .....

.....

DAYTIME TELEPHONE NO .....

I enclose a cheque/P.O. to the value of £17.65 (inc. VAT) for a lineage advertisement of no more than 30 words. Please make cheques payable to MCPC Ltd

QUANTA

HARDWARE

Care Electronics
0923 672102
Peripherals and accessories.
Thor International
010 4533 930305
010 4533 937544
Digital Precision
081-527 5493
Discs, interfaces, drives, keyboards, RAM expansions.
EEC Ltd
0753 888866
QLs, peripherals, accessories
Miracle Systems
0904 423986
Trump Card interface, modems, hard disc drive.
Strong Computer Systems
0267 231246
Sector Software
0772 454328
Peripherals.
S.P.E.M
Via Aosta 16, 10154 Torino, Italy.
+39 11 857924 Fax +39 11280009
QL disk interfaces, ram, eprom holders.
TF Services
071 724 9053
Repairs, spares, peripherals.
TK Computerware
0303 812801
Peripherals.

SERVICES

Adman Services
0952 255895
Joe Atkinson
36 Ranelagh Road, London W5 5RJ
Roms, mdvs and all spares.
Suredata
081-902 5218
Repairs.
QL World
(reader services)
Subscriptions
091 510 8787 (Lazerhold)
Microdrive Exchange
0772 454328 (Sector)
Recent back issues : write to editor
Quanta
User Group, support.
Hon. secretary, Phil Borman.
0933 460690
Xchange User Group
Freepost
Beckenham
Kent BR3 2BR

SOFTWARE

Athene Consultants
0705 511439
ARK Distribution
0983 79496
Archivist, Master Spy and other business and utility software.
Call Us What You Like
55 Greenan, Shaws Road, Belfast BT11 8LX. No telephone. Superspy.
Care Electronics
0923 672102
Tony Tebby software. Miracle hardware, peripherals, utilities.
CGH Services
0559384 574
Adventures. QLAF, DIY Toolkit.
Compware
0270 582301
Task Swopper, Mega Toolbox, Expert, System Shell, Speedscreen and others.
COWO Electronic
010 41 45 211478 (Switzerland)
QTop, Atari QL emulator, Thor support
Digital Precision
081-527 5493
Turbo Basic compiler, Desktop Publisher, Editor, Eye-Q Graphics. Sprite generator, Super Forth, Better Basic, Professional and Super Astrologer, Media Manager, Supercharge, Blocklands, Droidzone, Aracadia, Reversi, Backgammon, Master Bridge Tutor, C-M Emulator, C compiler, Lightning speed enhancer.
DJW Software
0256 881701
Homebanker
Dilwyn Jones Computing
0248 354023
Home Budget, Vision Mixer, QL Genealogist, Textidy, Spellbound and others.
Eidersoft
089283 2552
Games utilities, Ice. Pyramide.
ImaQLate Software
42 Albion Street, Broadstairs, Kent CT10 1NE.
Biograph.
Jochen Merz Software
010 49 203501274
QL/Atari Emulators, utility and games software.
Lear Data Systems
6 Southview Green, Bentley
Ipswich, Suffolk IP9 2DR
Qtris
Liberation Software
081 546 7795
Basic compiler (QLib) and utilities.
QFILE
Apartado 2110 p-1103 Lisboa Codex, MS-Link and Discopy. Portugal.
QL Supersoft
0256 475992
Spell, Numbers, Superdraw.
Qubbesoft PD
38 Brunwin Road, Rayne, Braintree, Essex CM7 5BU
Public domain software.
Sector Software
0772 454328
Flashback and other utilities.
SD Microsystems
0462 422897
Business software including General Ledger. Small Traders' Pack/Invoicer and Stock Accounting.
Software 87
33 Savernake Road, London NW3 2JU
Text 87 and updates.
Talent Software
0303 813883
Utilities, games.
TK Computerware
0303 812801
Add-ons, spares, software.
WD Software
0534 81392

DJC

Dilwyn Jones Computing

41 Bro Emrys, Tal-y-Bont,
Bangor, Gwynedd LL57 3YT
Tel: Bangor (0248) 354023

COCKTAILS WAITER

Christmas is coming, as we all know! If you keep drinks in the house over the festive season, why not let this program conjure up a cocktail or two when you hold a party or have friends and relatives round. A database of 400+ recipes is supplied as standard - further recipe sets are also available (see below). Needs at least 256k of memory to run. Tell it which ingredients you have handy and it will give you a list of cocktails you can make with those drinks! Great fun.

On disk £10.00 on 2 cartridges £12.00
MIX2 - hundreds of extra recipes £5.00
MIX3 - still more recipes, including many non-alcoholic drinks! £5.00

FLOPPY DISKS, ETC. . .

Make sure you have enough floppy disks, labels, storage boxes, etc to last the Christmas and New Year period. Contact DJC for any items you may need, and don't forget we can now accept telephone orders if you wish to pay by Visa/Access/Mastercard/Eurocard credit cards for quick delivery, subject to items being in stock.

3.5" unbranded floppy disks (DSDD)	45p each
(with labels) 20 or more	40p each
Standard 3.5" disk labels, roll of 100	£2.00
3.5" labels on tractor feed roll of 100	£2.50
5.25" DSDD unbranded disks	35p each
Box to hold 40 3.5" disks	£5.00
Box to hold 80 3.5" disks	£7.00
Posso box, holds 150 3.5" disks	£17.00
Disk box subject dividers, set of 20	£3.00
Monitor stand, for 12" or 14" monitors	£15.50
Small plastic box for 10 3.5" disks	£1.20
Microdrive cartridges	£2.50
Cartridge labels, roll of 100 labels	£2.00
Address labels on tractor fed roll 100	£2.00

PLEASE ADD POSTAGE AND PACKING AT THE RATES SHOWN ON PAGE 2 FOR THESE ITEMS

SLOWGOLD 2

SLOWGOLD 2 by Norman Dunbar and Phil Borman Slowgold 2 is a utility for slowing down programs running too fast on a Gold Card (although it can also be used on a standard QL if required). There are two versions of Slowgold - one interrupt driven routine for slowing down the whole computer and a new version that can selectively slow down individual programs when several are in use at the same time (which the Gold Card's SLUG command is not able to do). A bargain at only **£5.00!**

"THE CAT"

The Cat is a BASIC extension that prints to the screen or printer a tidy list of files on a disk or cartridge in columns to save the frustration of only being able to see a small number of files at a time on screen or printing several sheets of paper just for the one list of files. Installs itself as a SuperBASIC extension, so it stays even after a NEW, etc.

On disk ONLY £5.00 or on Cartridge ONLY £7.00

SCREEN SNATCHER

Capture screen displays from other programs on the QL. A short multitasking routine sits alongside other programs waiting for you to press a key combination you define, when it springs into action and saves a snapshot of the current display to disk, ramdisk ror microdrive, so that you can load it into another program for printing if required, or so that it can be included in a Vision Mixer presentation if required. We use it to include screen shots in our catalogue, for example.

On disk £10.00 on cartridge £12.00

QL GENEALOGIST Second Edition

The enhanced version of the very popular QL Genealogist program. Now with improved research reports, new Geography section, improved Notes facility, TREE outputs, compatible with version one data files, and many more minor improvements to many parts of the program. Includes a tutorial chapter in the manual to help get you going with the program. Upgrade terms available for buyers of the first version, and a conversion service is available from the author for those who currently use Archive to record family history who wish to move up to this program.

GENEALOGIST Second Edition (disk only)	**£30.00**
Standard Genealogist (while stocks last)	**£19.50**
Budget128k Genealogist on microdrive	**£12.00**
Budget 128k Genealogist on disk	**£10.00**

Archive _dbf file conversion service - ask!

PICTUREMASTER,

PICTUREMASTER, the screen making utility for use with out Vision Mixer programs, has been significantly upgraded. Please contact us for details and our upgrade terms.

COMING SOON IN THE NEW YEAR

Upgrades to Page Designer 2 and Image Processor (formerly sold by Sector Software). Contact us for full details and prices.

PAYMENTS - We are now able to accept payment by Mastercard, Access, Visa or Eurocard credit cards as well as the methods listed in our advert on page 2.